DASP Documentation

Hongzhiwei Technology (Shanghai) Co., Ltd.

2024-01-29

Contents:

Warning: Dear valued customer,

Hello! Due to recent changes in the Material Project database API, there is a high probability of errors occurring when using the TSC module in DASP 2022B to call the mp_api for stability assessments. We are currently conducting urgent adjustments and testing, and we plan to fix and enhance the TSC module in next version. We strongly recommend that you refrain from using the TSC module in DASP 2022B (dasp 2).

We sincerely apologize for any inconvenience and trouble this may have caused you. We deeply appreciate your understanding.

DASP 2022B has released, for more information, please refer to the simplified Chinese version of the manual.

1

1 Introduction

1.1 1.1 What is DASP

DASP (Defect and Dopant ab-initio Simulation Package) is a first-principles computational simulation package for calculating defect and dopant properties in semiconductors. According to the input crystal structure of semiconductor, based on the materials genome database and first-principles software package, DASP can automatically calculate and output the thermodynamic stability, defect (dopant) formation energies and transition energy levels of semiconductors, the defect (dopant) and carrier concentrations and Fermi levels, the photoluminescence spectra induced by the dominant defect (dopant), the capture cross-sections of carrier radiative and nonradiative, as well as minority carrier lifetimes.

Thermodynamic Stability Calculation (TSC): Firstly, the crystal structure is read in, and then the structural relaxation and total energy calculations are performed. Following, search for the competing secondary compounds in the material genome database (e.g. Materials Project), and combine the formation energy information of all the secondary compounds with the calculated formation energy of the host compound (equivalent inputs are used for DFT calculations), the chemical potential region that make the pure host stable can be determined, which can be act as the inputs for the judgment of thermodynamic stability and further defect (dopant) properties calculations.

Defect Energy Calculation (DEC): Construct an approximate cubic supercell based on the primitive cell configuration of the input structure. Then comprehensively consider various defect (dopant) configurations based on the crystal symmetry, following the supercells that contain defects (dopants) are generated and a first-principles software package is used to carry out structure optimization, electronic structure, and total energy calculation. According to the achieved results, predicting the possible charged states of defects (dopants), and calculating the properties of charged defects (dopants) further. Finally, based on the first principle calculation results and TSC module results, the formation energy and ionization energy level of defects are automatically calculated, and various error corrections are also calculated.

Defect Density Calculation (DDC): Predicting the concentration of various defects and impurities, Fermi level, and carrier density in semiconductors prepared under different chemical potentials and temperatures by reading the TSC and DEC module results, and giving their relationships with element chemical potential, and preparation and working temperature. The calculation results of the DDC module can be used to determine the dominant defects and impurities, and compare them with the concentrations measured by experiments, providing a quantitative basis for the regulation of defects, impurities, and carrier concentration.

Carrier Dynamics Calculation (CDC): For the critical defects determined by the DDC results, this module can calculate their excited-state carrier dynamics properties based on the Fermi level. Then the phonon spectrum of the defect/dopant supercell, electron-phonon coupling matrix and transition dipole moments between the defect/dopant states and VBM or CBM states will be calculated, based on which the radiative and non-radiative transition rates (carrier capture cross sections) and the lineshape of photoluminescence spectra induced by defects and dopants can be calculated. Combining the calculated transition rates and the equilibrium defect and carrier densities, the CDC module may also calculate the Shockley-Read-Hall recombination rates and the non-equilibrium carrier lifetime.

Fig 1. The framework of the DASP software, which is composed of four modules, TSC, DEC, DDC and CDC. The major functions of the four modules are shown in the boxes.

For any semiconductor, DASP software can calculate and give the following properties: thermodynamic stability, stable range of element chemical potential, defect (including impurities, the same below) formation energy and transition energy level, Fermi energy under various growth conditions, carrier and defect concentration, defect photoluminescence spectrum, carrier capture cross-section, radiative and nonradiative recombination rates, etc.

1.2 1.2 Calculation flow

(1) prepare Input files: `POSCAR` and `dasp.in`

(2) PREPARE: generate the parameters and input files required for VASP calculation.

(3) TSC: perform thermodynamic stability calculations on the target semiconductor, and determine the stable range of elemental chemical potentials.

(4) DEC: calculate the defect formation energy and transition energy level (ionization level).

(5) DDC: calculate Fermi level and concentrations of carrier and defects.

(6) CDC: calculate defect-induced photoluminescence spectra, carrier radiative, and nonradiative trapping rates.

Fig 2. The flowchart of DASP. Different colors represent the four modules. The dashed lines show the calculations that need to call external ab-initio DFT softwares.

In short, all calculations can be completed only need to execute the following five commands in sequence after preparing the two input files, `POSCAR` and `dasp.in` :

`dasp 1` (corresponding to PREPARE)
`dasp 2` (corresponding to TSC)
`dasp 3` (corresponding to DEC)
`dasp 4` (corresponding to DDC)
`dasp 5` (corresponding to CDC)

You can query the progress of the computing task through `tsc-state` and `dec-state` during TSC and DEC calculation, respectively.

1.3 1.3 Operating environment requirements

1.3.1 1.3.1 First-principle calculation software: VASP

The first-principles calculation software package, Vienna Ab initio Simulation Package (VASP), is needed to perform defect-related structural and electronic structure calculations for the DASP software. Therefore, the user needs to provide the compiled VASP executable file directory.

1.3.2 1.3.2 Materials Project database

During the installation of the DASP package, Pymatgen will be automatically installed through the `pip install` command.
During the execution of the program, the `Materials Project` database will be accessed through Pymatgen, so the user must register an account in `Materials Project` and obtain the API.
Since all calculations in this database use VASP's version 2003 pseudo-potential files, the user must correctly configure Pymatgen to match the 2003 version of the pseudo-potential library.
The specific operation steps are as follows:

1. Users can download the 2003 version of the POTCAR file by themselves.
2. Please refer to the pymatgen website for the next configuration:
https://pymatgen.org/installation.html, focusing on the following:

After installation, do:

```
pmg config -p <EXTRACTED_VASP_POTCAR> <MY_PSP>
```

In the above, <EXTRACTED_VASP_POTCAR> is the location of the directory that you extracted the downloaded VASP pseudopotential files. Typically, it has the following format:

```
- <EXTRACTED_VASP_POTCAR>
|- POT_GGA_PAW_PBE   # (must be version 2003)
||- Ac_s
|||-POTCAR
|||-...
```

or

```
- <EXTRACTED_VASP_POTCAR>
|- potpaw_PBE  # (must be version 2003)
||- Ac_s
|||-POTCAR
|||-...
```

and follow the instructions. If you have done it correctly, you should get a resources directory with the following directory structure:

```
- psp_resources
|- POT_GGA_PAW_PBE
||- POTCAR.Ac_s.gz
||- POTCAR.Ac.gz
||- POTCAR.Ag.gz
```

(continues on next page)

(continued from previous page)

```
...
|- POT_GGA_PAW_PW91
...
```

After generating the resources directory, you should add a VASP_PSP_DIR config variable pointing to the generated directory and you should then be able to generate POTCARs:

```
pmg config --add PMG_VASP_PSP_DIR <MY_PSP>
```

1.4 1.4 Scope of use

DASP can calculate the defect and impurity properties of semiconductors and insulators. Therefore, users need to check whether the target material has a band gap through the band structure before defect calculation. For some narrow-bandgap semiconductors, exchange-correlation approximations such as GGA and LDA may cause the bandgap to be underestimated or even disappear. In this case, hybrid functionals are required for calculation (see level=2 and 3 in the specific parameters section for details).

1.5 1.5 Bibliography

[1] Menglin Huang, Zhengneng Zheng, Zhenxing Dai, Xinjing Guo, Shanshan Wang, Lilai Jiang, Jinchen Wei, Shiyou Chen, DASP: Defect and Dopant ab-initio Simulation Package, Journal of Semiconductors, 4, 042101 (2022); doi: 10.1088/1674-4926/43/4/042101; https://arxiv.org/abs/2201.02079.

[2] Menglin Huang, Zenghua Cai, Shanshan Wang, Xin-Gao Gong, Su-Huai Wei, Shiyou Chen, More Se Vacancies in Sb2Se3 under Se-Rich Conditions: An Abnormal Behavior Induced by Defect-Correlation in Compensated Compound Semiconductors, Small, 17, 2102429 (2021).

[3] Menglin Huang, Shan-Shan Wang, Yu-Ning Wu, Shiyou Chen; Defect physics of ternary semiconductor ZnGeP2 with a high density of anion-cation antisites: A first-principles study, Physical Review Applied, 15, 024035 (2021).

[4] Jinchen Wei, Lilai Jiang, Menglin Huang, Yuning Wu, Shiyou Chen, Intrinsic Defect Limit to the Growth of Orthorhombic HfO2 and (Hf,Zr)O2 with Strong Ferroelectricity: First-Principles Insights, Advanced Functional Materials, 31, 2104913 (2021).

[5] Shiyou Chen, Aron Walsh, Xin-Gao Gong, Su-Huai Wei, Classification of Lattice Defects in the Kesterite Cu2ZnSnS4 and Cu2ZnSnSe4 Earth-Abundant Solar Cell Absorbers, Advanced Materials, 25, 1522 (2013).

2

2 Modules

2.1 2.1 PREPARE Module

There are two main input files in DASP: `POSCAR` and `dasp.in` , where `dasp.in` is the main parameter control file.

Before calculation, PREPARE module would check if the input parameters in the `dasp.in` are reasonable first. If it is, the next calculation begins and marks out that the preparation work has been done in the `1prepare.out` .

If it isn' t, the information of mistakes and warnings will be written in the `1prepare.out` and program terminated. Users need to revise the input parameters based on the warnings and restart program.

Since DASP calls VASP to carry out the structure and electronic structure calculation, thus DASP will automatically generate the input files required for VASP calculation according to the parameters in `dasp.in` : `POSCAR` (crystal structure), `INCAR` (calculation parameters), `KPOINTS` (k-points setting), `POTCAR` (pseudo-potential file), and submission script.

- **POSCAR**

  According to the `POSCAR` that user input, DASP will generate a supercell that is nearly cubic within a given atoms number through the self-developed algorithm "nearly cubic supercell" . Then, fix the lattice constant and optimize the positions of all atoms in the supercell to obtain `POSCAR_final` file.

- **KPOINTS**
  For the calculations of supercells with defects, the single k point is used in DASP, namely only Gamma point is included in the `KPOINTS` file.
- **INCAR**
  DASP will generate two INCAR files, one is `INCAR-relax` for the structural optimization, ant the other is `INCAR-static` used in static calculation. The default generated is commonly used parameters. They can be modified after the PREPARE module runs if users need to modify them. According to the parameters in the `dasp.in` , DASP can adopt three different calculation levels (see input parameter level=1/2/3 for details). For level=2 or 3, the hybrid functional will be used. DASP can automatically determine the proportion of exchanged parts in hybrid functional by the experimental band gap that user set in `dasp.in` , and the matched proportion is written into INCAR.
- **POTCAR**
  According to the path of pseudo-potential file provided by the user, DASP will automatically generate the `POTCAR` required for calculation.
- **Submission script**
  DASP will automatically generate a submission script file based on the system name, queue name, nodes, cores and the path of VASP written by the user in `dasp.in` .

As mentioned above, PREPARE module will automatically call VASP to optimize the structure and determine the proportion of the exchange part in the hybrid functional. If user specifies the Lany-Zunger correction scheme in `dasp.in` (see correction in the introduction of input parameter for details), this scheme needs to know the Madelung constant corresponding to the supercell. According to the generated supercell, the PREPARE module will call VASP to automatically calculate the Madelung constant and write it into `dasp.in` . The Madelung constant here is only related to the lattice of the supercell.

All details of the above generated files and calculation process will be written into `1prepare.out` . The information on status, results and errors can be queried in the file. After this module runs successfully, it will be noted the complete information at the end of the `1prepare.out` , which will be detected when the subsequent TSC module starts.

2.2 2.2 TSC Module

The TSC module is for calculating the chemical potential ranges through considering the influences of all the competing secondary compounds that can limit the thermodynamic phase stability of the host compound semiconductor, in which the chemical potential will be used as the input for the subsequent calculation on DEC and DDC modules.

Whether a compound semiconductor is stable depends on its formation energy. If it has more advantages in formation energy than its competitive pure phase and hetero-phase compounds, the compound can be thermodynamically stable. The judgment process is as follows.

- **Judgement of thermodynamic stability of semiconductors**
  A semiconductor if it is thermodynamically stable (does not decompose into simple substances or other hetero-phase, and can synthesize pure phase sample) needs to match the following three conditions:

(1) The formation of the target compound reaches thermodynamic equilibrium:
Under the equilibrium condition, the formation and decomposition of the target compound are in

dynamic equilibrium. For a compound $A_kB_lC_mD_n$, the weighted sum of the chemical potentials μ of its component elements should be equal to the formation energy E^f of the compound,

$$k\mu_A + l\mu_B + m\mu_C + n\mu_D = E^f(A_kB_lC_mD_n) \tag{2.1}$$

(2) The formation of the competing secondary compounds concerning the host compound cannot be carried out:

For any hetero-phase compound $A_{k'}B_{l'}D_{n'}$, the chemical potential of each element and the formation energy of hetero-phase compound need to meet the following inequality:

$$k'\mu_A + l'\mu_B + n'\mu_D < E^f(A_{k'}B_{l'}D_{n'}) \tag{2.2}$$

(3) The simple substances of the constituent elements of the host compound will not form:

In order to avoid the formation of a simple substances, the chemical potential of each element satisfies the following inequality:

$$\mu_A < 0 \ , \ \mu_B < 0 \ , \ \mu_C < 0 \ , \ \mu_D < 0 \tag{2.3}$$

After considering all competing secondary compounds, the host compound can be thermodynamically stable if the chemical potential of each element can exist in a range that meets all the above constraints. Otherwise, the compound semiconductor is unstable and will be decomposed into simple substances or other hetero-phases, and the synthesized sample cannot guarantee a pure single-phase.

The result of thermodynamic stability given by the above conditions and processes actually has an equivalent effect to the energy above hull of compound in judging thermodynamic stability. If the energy above hull is positive, the compound is unstable, and there is no range of element chemical potential, which meets all the above conditions. If the energy above hull is negative, the compound is stable, and there is a range of element chemical potential, which meets all the above conditions.

- **Two steps of thermodynamic stability calculation**

  According to the above discussions, the formation energy of all competing secondary compounds is required to calculate the thermodynamic stability and the stable range of element chemical potential of compound semiconductors. For binary, ternary, quaternary, quinary and even more compounds, there are many possible competing secondary compounds, which all need to be taken into account in the calculation. If not fully considered, some unstable compounds will be predicted to be stable. Therefore, full consideration of all possible secondary compounds is critical to the accurate calculation of thermodynamic stability and elemental chemical potential range. In order to consider all secondary compounds as fully as possible, DASP will visit the Materials Project (MP) database to search for all the compounds that are composed of the component elements, and quickly determines the critical competing compounds by their formation energy. Then, for the host and critical competing compounds, TSC can calculate their formation energies with higher accuracy to ensure that the calculated range of the chemical potentials is accurate. It is divided into two steps:

- **First step**

  DASP will visit the materials genome database, such as Materials Project (MP) database, to search for all the compounds that are composed of the component elements, then obtained the total energy and structure of these compounds, and generate VASP calculation input files in which the parameters consistent with the MP database: `INCAR` , `KPOINTS` , `POTCAR` , `POSCAR` (copied from the files provided by user). Meanwhile, TSC will also perform a VASP calculation for the total energy and formation energy of the host compound (the calculated energy can be compared to that in MP database directly). The process of calculation is the same as that of the MP database, which is divided into twice structural optimization and one static calculation, which are `relaxation1` , `relaxation2` , and `static` under "TSC/ host homonymous directory".

With the formation energies of the host compound and all the competing compounds, TSC can solve the thermodynamic constraint equations and inequations to predict whether the host compound is stable, and determine the critical competing compounds that limit the stable chemical potential region. The structure and total energy of the primitive cell of the host compound are only needed to calculate at this stage, and GGA-PBE exchange-correlation is used leading to a small amount of calculation. A large number of data of secondary compounds are directly from the MP database without calculation. Therefore, all the competing secondary compounds can be considered quickly and fully to determine the critical competing compounds.

- **Second step**

  For the host and critical competing compounds, the unified VASP parameters and input files generated by the PREPARE module are adopted, `INCAR` and `POTCAR` , and `KPOINTS` and `POSCAR` (for the host compound, copied from the file provided by the user, for the critical competing compounds, download from the MP database) automatically generated by the TSC module, the energy and formation energy of the host and critical competing compounds are recalculated. In order to make the calculation quickly, only static calculations are made for the host and critical competing compounds, and the directory is located in "TSC/ host or secondary compounds " `static_recalc` . Then according to the formation energies of the host and critical competing compounds, TSC resolves the thermodynamic constraint equations and inequations to calculate the stable chemical potential region of each element as the input of the subsequent DEC and DDC modules.

Based on the formation energies of the host and critical competing compounds, the TSC module also automatically calculates and outputs the energy above hull (ev/atom) and the most possible decomposition path of the host compound. The energy above hull also shows the thermodynamic stability of the compound with respect to the phase separation into other competing compounds. The more negative the value, the more stable the compound is, and the more positive, the more unstable it is conversely. So TSC can also be used for high-throughput and accurate calculation of the thermodynamic stability of new compounds.

For the ternary and quaternary compound semiconductors, TSC module can draw two-dimensional and three-dimensional stable region phase diagrams in the chemical potential space. Please refer to the Examples for details.

All details of TSC module operation, results and errors are output in `2tsc.out` , the status of this module can be queried in this file. After this module runs successfully, it will be noted the complete information at the end of the `2tsc.out` , which will be detected when the subsequent DEC module starts.

2.3 2.3 DEC Module

According to the supercell model, the formation energy and transition energy level (ionization energy level) of defects and dopants can be calculated by calling ab-initio software in the DEC module. This is also the main result since the 1990s given in most first principle research about defects and dopants.

The DEC module will generate a series of configurations with defects and dopants based on the parameters set by the user in `dasp.in` and the supercell produced by PREPARE module. Then, based on the VASP input file generated by the PREPARE module and the chemical potential calculated by the TSC module, the DEC module will call ab-initio software VASP to calculate the structure and electronic structure of defects and dopants. According to the result of calculations, the formation energy and transition energy level of defects and dopants will be calculated, and output the figure.

Before generating the structure, the DEC module will detect the parameters in `dasp.in` whether reasonable. If it is, start the next calculation, and output the running status information in real-time in `3dec.out` . If it isn' t, relevant error and warning information will be output to `3dec.out` file, and the program terminates. Users need to modify relevant parameters according to the error information and restart.

The workflow of DEC module is divided into the following six steps: generating neutral defects, automatically submitting tasks to calculate neutral defects, generating charged defects, submitting tasks to calculate charged defects, calculating the defect formation energy with different charge states under different chemical potentials and Fermi levels, and outputting figures.

- **generating neutral defects**
  The neutral defects include vacancy, antisite, and interstitial. The supercell structure `POSCAR_final` is generated by using PREPARE module, the vacancy and antisite on inequivalent sites are produced based on the crystal symmetry, meanwhile, the interstitial is obtained by randomly scattering at the position far away from atoms. After the defect configuration is generated, the DEC module will put the VASP input files, such as INCAR, KPOINTS, POTCAR, and submission script, generated by the PREPARE module into the directory of each defect. For the dopants, there are only two configurations: antisite (substitution) and interstitial.
- **generating charged defects**
  The DEC module will create the corresponding calculation directory of charged defects according to the results of neutral defects (the occupied state of the eigenvalue of neutral defects). If the calculation of neutral defects fails or does not converge, there will be no charged defects generated.
- **automatically submitting jobs**
  DEC module will automatically call VASP to carry out structural optimization and static calculation for all neutral and charged defects and detect whether the calculation is successful and converged. The command `dec-state` used under the dec directory can view the status of all jobs to be calculated at any time, including converged, does not converge, error, running, queuing, not submitted, etc. After the DEC module runs, the user can enter the corresponding directory to modify the INCAR for tasks that do not converge and have errors, and write the path of this directory into `redo.in` under the dec directory and re-execute the DEC module.
- **formation energy calculation**
  The formation energy of a point defect in the charge state q can be calculated as,

$$E_f = E_{tot(defect)} - E_{tot(host)} - \Sigma_i n_i(\mu_i + E_i) + q(E_F + E_{VBM}) + E_{corr} \tag{2.4}$$

  where E_f is the formation energy, $E_{tot(defect)}$ and $E_{tot(host)}$ are the total energies of the supercells with and without a defect (dopant). n_i is the number of atoms of element i remove from ($n_i < 0$) or added to ($n_i > 0$) the supercell to form the defect, and μ_i is the elemental chemical potential referenced to the total energy E_i of the pure solid/gas elementary phase. q is the number of electrons, transferred from the supercell to the reservoirs in forming the defect cell. E_F is the Fermi level referenced to the eigenvalue of the valence band maximum (VBM) level of the bulk supercell. E_{corr} is the correction that accounts for the spurious interaction caused by finite supercell size and periodic boundary conditions.

  The DEC module will read the output of the first principle calculation, calculate the formation energy of each charged state of each defect according to the formula, and automatically calculate the correction value according to the correction method set by the user. If the defect calculation fails or does not converge, the formation energy will not be calculated.
- **outputting figures**
  The DEC module can output the figure of the defect formation energy changing with the Fermi

level according to the result of the formation energy, including the data in format dat: `p1.dat, p2.dat, ...` , and pictures with png format: `p1.png, p2.png, ...` (the integer represents the number of points in the chemical potential space given by the TSC module), as well as the data and picture of the transition energy level: `tl.dat` and `tl.png` . Please draw the picture according to the instructions in the document.

- **distorted defect structure calculation**
  The DEC module can automatically generate the distorted defect structure based on the results of the original structure. It is worth noting that specifying the defect that will produce the distorted structure must be after the calculation of the original defect structure is completed, and running the DEC module again. For the defects that have not completed the initial structure calculation, the distorted structure will not be generated.

All details of the above generated files and calculation process will be written into `3dec.out` . The information on status, results and errors can be queried in the file. After this module runs successfully, it will be noted the complete information at the end of the `3dec.out` , which will be detected when the subsequent DDC module starts.

2.4 2.4 DDC Module

Defect concentration is an important parameter for the performance optimization and simulation design of semiconductor materials and devices. Experimentally, the growth conditions of the material can be controlled to inhibit the formation of harmful defects or promote favorable defects forms, to achieve the purpose of optimizing the performance of devices. In terms of calculation, a further prediction of the relationship of the defect concentration verse the growth conditions, such as chemical potential, on the basis of the defect formation energy and transition energy level results, can provide a more clear and quantitative reference for the regulation of semiconductor performance.
The work of DDC module is to read the results of formation energy and transition energy level under various chemical potentials calculated by TSC and DEC modules before, and self consistently solve the Fermi level, defect concentration, and carrier concentration of the system at a certain growth and working temperature according to the charge neutrality equation.

For a defect in the charge state q, the concentration under equilibrium conditions can be represented as,

$$N(,q) = N_{sites} g_q exp[-E_f/k_B T] \tag{2.5}$$

Where N_{sites} is the number of sites that defects can be incorporated per volume, g_q is the degeneracy factor that equals to the number of possible electron configurations for different charge states, E_f is the defect formation energy. All the ionized defects in the charge state q 0 produce carriers. The positively charged donor defects with q>0 produce electron carriers, and their summed charge is $\sum_{\alpha,q>0}[q * n(\alpha,q)]$; while the negatively charged acceptor defects with q<0 produce hole carriers, and their summed charge is $\sum_{\alpha,q<0}[(-q) * n(\alpha,q)]$. The final densities of electron and hole carriers are contributed by both the thermal excitation and the ionization of all these defects (dopants). The equilibrium-state Fermi level can be calculated through solving the charge neutrality equation,

$$n_0 + \sum_{\alpha,q<0}[(-q) * n(\alpha,q)] = p_0 + \sum_{\alpha,q>0}[q * n(\alpha,q)] \tag{2.6}$$

where $\sum_{\alpha,q<0}[(-q) * n(\alpha,q)]$ and $\sum_{\alpha,q>0}[q * n(\alpha,q)]$ are the summed charges of negatively charged defects and positively charged defects, weighted by the charge q. n_0 and p_0 are free carrier densities, which can be

defined as,

$$n_0 = \int_{\varepsilon_c}^{+\infty} g(E)f(E)dE \tag{2.7}$$

$$p_0 = \int_{-\infty}^{\varepsilon_v} g(E)(1 - f(E))dE \tag{2.8}$$

where $g(E)$ is the density of states (DOS), and $f(E)$ is the Fermi-Dirac distribution function.

The semiconductors are usually grown or synthesized at a high temperature and then go through a rapid annealing process to a lower working (measuring) temperature. Therefore, the defects are usually formed at the high temperature, and then it can be assumed that the atomic structure will not change and new defects will not form in the subsequent rapid annealing process. But the densities of different charge states will redistribute during the rapid annealing. The DDC module is developed in accordance with such fabrication process, i.e. self-consistently solving the charge neutrality equation at a high growth temperature firstly, and then the Fermi level and the densities of each defect in different charge states can be obtained at the high temperature. Afterward, self-consistently solves the charge neutrality equation again at the lower working temperature, but now the formula for the concentrations calculation of defects in different charge states should be modified. In the second self-consistently solves step, the density summation for each defect over all charge states is fixed, and the density of each charge state will undergo a redistribution according to the Fermi-Dirac distribution. Then a new Fermi level can be obtained at the working temperature, and the redistributed defect densities and carrier densities can be calculated.

DDC module can calculate the Fermi level, carrier densities, and concentrations of differently charged defects under various chemical potentials, growth and working temperature based on the results and output files of TSC and DEC. It mainly includes the following steps:

- **summary of formation energies**
  Based on the defects calculation in DEC, DDC will automatically search the data, such as formation energy and transition energy level, that output under each defect directory, then summarize and output them in the file `DefectParams.txt` .
- **self-consistent calculation of Fermi level**
  DDC self-consistently solves the charge-neutrality equation to determine the Fermi level at growth and working temperature respectively, and the results will be written into `4ddc.out` and `Fermi.dat` .
- **calculation of defect and carrier concentration**
  Based on the Fermi level, the corresponding carrier densities and differently charged defect concentration can be calculated, and output the data file `Carrier.dat` and `Defect_charge.dat` , as well as the png image `density.png` .

All details of the above generated files and calculation process will be written into `4ddc.out` file. Relevant status, results, and error information can be queried in this file.

2.5 2.5 CDC Module

The CDC can calculate the following properties: (1) radiative carrier capture coefficient of defects, (2) lineshape of photoluminescence spectra, and (3) phonon-assisted nonradiative carrier capture coefficient (cross section).

(1) radiative carrier capture coefficient

Assuming that defect A has different charge states q=0 and q=+1, and the (0/+) transition energy level located in the band gap, so the defect A^+ with +1 charge state can capture electrons from conduction band minimum (CBM) and convert to A^0 . If this process is the radiative transition (releasing photons), its capture coefficient can be expressed as:

$$C_n = f_{spin} V_{supercell} \frac{n_r e^2}{3m^2{}_0 c^{32}} |\langle \psi_i | p | \psi_f \rangle|^2 E_{opt} \tag{2.9}$$

where f is the Sommerfeld factor used to express the Coulomb interaction between the charged defect before capture and the trapped carrier. As it is Coulomb attraction for A^+ to capture an electron, the order of magnitude of f is about 5 to 10. Conversely, when A^0 captures a hole becomes A^+ , f will be 1 due to the neutral defect neither attracting nor repelling positive and negative charges. $_{spin}$ is the selection rule of spin, $V_{supercell}$ is the volume of the supercell, n_r represents the refractive index, m is the mass of free electrons, and $\langle \psi_i | p | \psi_f \rangle$ is the momentum matrix element, which needs to modify the VASP source program and recompile to output. E_{opt} is the optical transition level of defects, which can be obtained by subtracting the lattice relaxation energy from the thermodynamic transition level of defects.

CDC module can call VASP automatically in batches to do first principle calculation based on the directory and output of DEC module (such as structure, transition energy level, etc.) to obtain the radiative capture coefficient in $cm^3 s^{-1}$. Based on radiative capture coefficient C_n , it is necessary to multiply the coefficient with the real defect concentration N_D (refer to the output of DDC module) to obtain the capture rate r , and then take the reciprocal to obtain the lifetime of radiative transition:

$$= \frac{1}{r} = \frac{1}{C_n N_D} \tag{2.10}$$

(2) lineshape of photoluminescence spectra

Steady-state photoluminescence spectroscopy is one of the important experimental measurements to characterize defects. Previous theoretical studies often only compared the calculated transition energy level with the PL peak to speculate the possible defect configuration, while the CDC module of DASP can simulate the shape of PL spectrum, so as to provide physical quantities such as the peak position, FWHM, zero phonon line, and Huang-Rhys factor of PL spectrum for comparison with experiments. The simulation of PL spectrum is mainly dependent on the calculation of "spectral function", that is, the intensity is written as a function of the radiant photon energy :

$$I() = N^3 \sum_m p_m \sum_n |\langle _{im} | _{fn} \rangle|^2 (E_{ZPL} + E_{im} - E_{fn} -) \tag{2.11}$$

Where N is the normalization factor, $_{im}$ and $_{fn}$ are the vibrational wavefunctions of the initial and final states, and E_{im} and E_{fn} are the corresponding eigenvalues. E_{ZPL} is the zero phonon line of the transition process. In order to evaluate the accurate vibrational wavefunction and the eigenvalues, one-dimensional configuration coordinate diagram is used in CDC to plot the potential energy surfaces, and the steady-state Schrodinger equation can be solved to obtain the numerical solutions of the wavefunction and eigenvalues.

CDC module can call VASP automatically in batches to do first principle calculation based on the directory and output of DEC module (such as structure, transition energy level, etc.) to calculate the configuration diagram, wavefunction, eigenvalues, and numerically solve the above overlapping integral, and finally output the intensity with the radiation photon energy , that is PL spectrum.

3

3 Input

3.1 3.1 The template of dasp.in

The parameters in the input file `dasp.in` can be divided into five categories as follows:

- 1. **bool**
  - Only the first character of this type parameter is read, if the first character is T/t, it is regarded as True; otherwise, it is regarded as False.
- 2. **Strings**
  - All input characters of this type parameter are read and processed in str format. If a number is entered, it will be converted to strings.
- 3. **integer**
  - All input characters of this type parameter are read and processed in int format. The input must be an integer, decimals are not supported.
- 4. **float**
  - All input characters of this type parameter are read and processed in int or float format. The input format supports integer or decimal.
- 5. **list**
  - All input characters of this type parameter are read, and separated by `space` into one or more values. The type of each value is one of I-IV.

The following is the `dasp.in` required to calculate the intrinsic defect of a material. Only necessary parameters are included in the following example. For the unset parameters see sections 3.2 to 3.5 for details. Users can adjust the parameters by themselves.

```
############## Job Scheduling ##############
cluster = SLURM     # (job scheduling system)
node_number = 2         # (number of node)
```

(continues on next page)

(continued from previous page)

```
core_per_node = 32      # (core per node)
queue = batch           # (name of queue/partition)
max_time = 24:00:00        # (maximum time for a single DFT calculation)
vasp_path_dec = /opt/vasp.5.4.4/bin/vasp_gam   #  (path of VASP)
vasp_path_tsc = /opt/vasp.5.4.4/bin/vasp_std
job_name = submit_job    # (name of script)
potcar_path = /opt/POT/potpaw_PBE    # (path of pseudopotentials)
max_job = 5

############## TSC Module ##############
database_api = ******************** # (str-list type)

############## DEC Module ##############
level = 2   # (level=1: PBE+PBE; level=2: PBE+HSE; level=3: HSE+HSE)
min_atom = 190
max_atom = 240
intrinsic = T   # (default: T)
correction = FNV   # (default: none)
epsilon = 10.3
Eg_real = 1.45   # (experimental band gap)

############## DDC Module ##############
ddc_temperature = 1000 300
ddc_mass = 0.09 0.84
```

3.2 3.2 System management

3.2.1 3.2.1 cluster

cluster indicates the system name, PBS and SLURM systems are supported.

cluster is **str** type, no default, and **required**.

Example of *cluster* setting:

```
# Default: No default
cluster =

# PBS system
cluster = PBS/pbs/... # Both uppercase and lowercase are valid parameters

# SLURM system
cluster = SLURM/slurm/... # Both uppercase and lowercase are valid parameters
```

3.2.2 3.2.2 node_number

node_number indicates the number of nodes used in a single task.

node_number is **int** type, no default, and **required**.

Example of *node_number* setting:

```
# Default: No default
node_number =

# Do not exceed the total number of nodes in the queue

# Example, please revise by the self
node_number = 1
```

3.2.3 3.2.3 core_per_node

core_per_node indicates the number of cores used in each node.

core_per_node is **int** type, no default, and **required**.

Example of *core_per_node* setting:

```
# Default: No default
core_per_node =

# Do not exceed the total number of cores of a single node in the queue

# Example, please revise by the self
core_per_node = 24
```

3.2.4 3.2.4 queue

queue indicates the queue name used for calculation.

queue is **str** type, no default, and **required**.

Example of *queue* setting:

```
# Default: No default
queue =

# Example, please revise by the self
queue = normal
```

3.2.5 3.2.5 max_time

max_time indicates the maximum time of a single task, which will be forcibly terminated by the queue if the set time is exceeded.

max_time is **str** type, no default, and **required**.

Example of *max_time* setting:

```
# Default: No default
max_time =

# format: HH(H...):MM:SS, and HH >= 0, 0 <= MM < 60, 0 <= SS < 60

# Example, please revise by the self
max_time = 24:00:00
```

3.2.6 3.2.6 vasp_path_dec, vasp_path_tsc, vasp_path_cdc

vasp_path_tsc , *vasp_path_dec* , and *vasp_path_cdc* indicate the path of VASP invoked using in the DASP-TSC, DASP-DEC, and DASP-CDC modules, respectively.

vasp_path_tsc , *vasp_path_dec* , and *vasp_path_cdc* are **str** type, no default, and **required**.

Example of *vasp_path_tsc* , *vasp_path_dec* , and *vasp_path_cdc* setting:

```
# Default: No default
vasp_path_tsc =
vasp_path_dec =
vasp_path_cdc =

# VASP must be installed in this path

# Example, please revise by the self
vasp_path_tsc = /opt/vasp5.4.4/vasp_std
vasp_path_dec = /opt/vasp5.4.4/vasp_gam
# vasp_path_dec = /opt/vasp5.4.4/vasp_std
vasp_path_dec = /opt/vasp_optics/vasp_gam
```

3.2.7 3.2.7 job_name

job_name indicates the name of the submission script.

job_name is **str** type, default is submit_job, and **optional**.

Example of *job_name* setting:

```
# Default:
job_name = submit_job

# Illegal characters: " ", "/", "?", "*", "$", "&", "(", ")"

# Example, please revise by the self
```

(continues on next page)

(continued from previous page)

```
job_name = job.pbs
# job_name = job.sh
```

3.2.8 3.2.8 potcar_path

potcar_path indicates the path to the folder containing pseudopotential files `POTCAR` .

potcar_path is **str** type, no default, and **required**.

Example of *potcar_path* setting:

```
# Default: No default
potcar_path =

# Please ensure that the pseudopotential folder exists under this path, and can be read␣
↪and write.

# Example, please revise by the self
potcar_path = /home/POT/potpaw_PBE
```

Note: Please ensure that the folder exists and has read and write permissions. Please make sure that there are not multiple identical pseudopotential files under this folder.

3.2.9 3.2.9 max_job

max_job indicates the maximum number of tasks allowed to run at the same time.

max_job is **int** type, default is 5, and **optional**.

Example of *max_job* setting:

```
# Default:
max_job = 5

# Example, please revise by the self
max_job = 3
```

3.3 3.3 TSC parameters

3.3.1 3.3.1 database_api

database_api indicates the key of application programming interface (API) of the Materials Project database, which is used to retrieve the information required for calculation.

database_api is **str** type, no default, and **required**.

Example of *database_api* setting:

```
# Default: No default
database_api =
```

3.3.2 3.3.2 key_phases_recalc

key_phases_recalc specifies whether TSC module calculates chemical potential with parameters consistent with DEC module. If set to *False* , the TSC module only calls the total energy on the MP database to judge the stability of the host compound without any DFT calculation of the secondary compounds.

key_phases_recalc is **bool** type, default is True, and **optional**.

Example of *key_phases_recalc* setting:

```
# Default:
key_phases_recalc = True

# Example, please revise by the self
key_phases_recalc = False
```

3.3.3 3.3.3 excluded_phase

excluded_phase specifies the secondary compounds to be excluded when analyzing the stability of the host compound. Multiple hetero-phases to be excluded can be set, and the names are separated by spaces.

excluded_phase is **list** type, the strings should be separated by space. The parameter has no default and is **optional**.

Example of *excluded_phase* setting:

```
# Default: No default
excluded_phase =

# Example, please revise by the self
excluded_phase = Zn(GaO2)2 Zn2InGaO5
```

3.3.4 3.3.4 axis_element_x

axis_element_x indicates the element chemical potential corresponding to the X axis in the two-dimensional stable region phase diagram of the host compound. It is only valid for ternary or quaternary semiconductors now.

axis_element_x is **str** type. The parameter has no default and is **optional**.

Example of *axis_element_x* setting:

```
# Default: No default
axis_element_x =

# Example, please revise by the self
axis_element_x = Cu
```

3.3.5 3.3.5 axis_element_y

axis_element_y indicates the element chemical potential corresponding to the Y axis in the two-dimensional stable region phase diagram of the host compound. It is only valid for ternary or quaternary semiconductors now.

axis_element_y is **str** type. The parameter has no default and is **optional**.

Example of *axis_element_y* setting:

```
# Default: No default
axis_element_y =

# Example, please revise by the self
axis_element_y = Zn
```

3.3.6 3.3.6 mid_element

mid_element indicates the element whose chemical potential is treated as intermediate variable in the two-dimensional stable region phase diagram of the host compound.

mid_element is **str** type. The parameter has no default and is **optional**.

Example of *mid_element* setting:

```
# Default: No default
mid_element =

# Example, please revise by the self
mid_element = Sn
```

3.3.7 3.3.7 fixed_chem_potential

fixed_chem_potential indicates the element whose chemical potential is fixed when drawing the two-dimensional stable region phase diagram of quaternary compounds, which is currently only valid for quaternary semiconductors. If the chemical potential exceeds the stable region after applying an offset, the module will automatically select a suitable value near the boundary of the stable region.

fixed_chem_potential is **str** type. Default: Automatically select the element according to the material, and the offset of chemical potential from the average value is 0 by default. Optional: the element name and the offset of its chemical potential from its average value.

Example of *fixed_chem_potential* setting:

```
# Default: Determinate by the program automatically

# Example, please revise by the self
fixed_chem_potential = Se:-0.2
# fixed_chem_potential = Se:-0.5
```

3.3.8 3.3.8 plot_2d

plot_2d specifies whether to output a two-dimensional stable region phase diagram for the host compound, only valid for ternary or quaternary semiconductors currently.

plot_2d is **bool** type. The default is Ture for ternary or quaternary semiconductors, and it is **optional**.

Example of *plot_2d* setting:

```
# Default:
plot_2d = True (ternary or quaternary compounds)
plot_2d = False (binary or quaternary compounds)

# Example, please revise by the self
plot_2d = False
```

3.3.9 3.3.9 plot_3d

plot_3d specifies whether to output a three-dimensional stable region phase diagram for the host compound, only valid for quaternary semiconductors currently.

plot_3d is **bool** type. The default is False, and it is **optional**.

Example of *plot_3d* setting:

```
# Default:
plot_3d = False

# Example, please revise by the self
plot_3d = True
```

3.3.10 3.3.10 tsc_only

tsc_only specifies whether to run the TSC module alone to analyze the stability of the host compound quickly. If set to Ture, the *potcar_path* must be set at the same time.

tsc_only is **bool** ype. The default is False, and it is **optional**.

Example of *tsc_only* setting:

```
# Default:
tsc_only = False

# Example, please revise by the self
tsc_only = True
```

3.4 3.4 DEC parameters

3.4.1 3.4.1 level

level indicates the method to calculate total energy within DASP, 1: PBE+PBE, 2: PBE+HSE, 3: HSE+HSE.

level is **int** type. The default is 1, and it is **optional**.

Example of *level* setting:

```
# Default:
level = 1

#  level = 1 represents that PBE is used for structure optimization and total energy␣
↪calculation. The level must be 1,2 or 3.

# Example, please revise by the self
level = 2     # level = 2 represents that PBE is used for structure optimization, and␣
↪total energy calculates by HSE.
# level = 1/2/3
```

3.4.2 3.4.2 min_atom

min_atom indicates the minimum number of atoms of supercells used for defects calculation.

min_atom is **int** type. The default is 64, and it is **optional**.

Example of *min_atom* setting:

```
# Default:
min_atom = 64

# num_r is the number of atoms in the refined cell, and there should be a multiple m to␣
↪make min_ atom <= m * num_ r <= max_ atom.

# Example, please revise by the self.
min_atom = 96
```

3.4.3 3.4.3 max_atom

max_atom indicates the maximum number of atoms of supercells used for defects calculation.

max_atom is **int** type. The default is 300, and it is **optional**.

Example of *max_atom* setting:

```
# Default:
max_atom = 300

# num_r is the number of atoms in the refined cell, and there should be a multiple m to␣
↪make min_ atom <= m * num_ r <= max_ atom.

```

(continues on next page)

(continued from previous page)

```
# Example, please revise by the self.
max_atom = 96
```

3.4.4 3.4.4 intrinsic

intrinsic indicates whether to carry out the intrinsic defects calculation.

intrinsic is **bool** type. The default is True, and it is **optional**.

Example of *intrinsic* setting:

```
# Default:
intrinsic = T

# Example, please revise by the self.
intrinsic = F
```

3.4.5 3.4.5 vacancy

vacancy indicates whether to calculate the vacancies.

vacancy is **bool** type. The default is True, and it is **optional**.

Example of *vacancy* setting:

```
# Default:
vacancy = T

# Example, please revise by the self.
vacancy = F
```

3.4.6 3.4.6 antisite

antisite indicates whether to calculate the antisite defects.

antisite is **bool** type. The default is True, and it is **optional**.

Example of *antisite* setting:

```
# Default:
antisite = T

# Example, please revise by the self.
antisite = F
```

3.4.7 3.4.7 interstitial

interstitial indicates whether to calculate the interstitial defects.

interstitial is **bool** type. The default is True, and it is **optional**.

Example of *interstitial* setting:

```
# Default:
interstitial = T

# Example, please revise by the self.
interstitial = F
```

3.4.8 3.4.8 doping

doping indicates whether to calculate the dopants.

doping is **bool** type. The default is False, and it is **optional**.

Example of *doping* setting:

```
# Default:
doping = F

# If there is a dopant, set doping = T and the parameter impurity must also be set.
# doping determines whether to generate the pseudopotential files of the doped element.

# Example, please revise by the self.
doping = T
```

3.4.9 3.4.9 impurity

impurity indicates the doped element.

impurity is **str** type. No default, and **optional**. **(It is required when doping = T)**

Example of *impurity* setting:

```
# Default: No default
impurity =

# Valid only when doping = T, and the value of impurity setting must be the discovered␣
↪element.

# Example, please revise by the self.
impurity = H
```

3.4.10 3.4.10 substitution_doping

substitution_doping indicates whether to calculate substitution defects when there is a dopant.

substitution_doping is **bool** type. The default is True, and it is **optional**.

Example of *substitution_doping* setting:

```
# Default:
substitution_doping = T

# Example, please revise by the self.
substitution_doping = F
```

3.4.11 3.4.11 interstitial_doping

interstitial_doping indicates whether to calculate interstitial defects when there is a dopant.

interstitial_doping is **bool** type. The default is True, and it is **optional**.

Example of *interstitial_doping* setting:

```
# Default:
interstitial_doping = T

# Example, please revise by the self.
interstitial_doping = F
```

3.4.12 3.4.12 num_inter

num_inter indicates the number of generated interstitial defects (intrinsic or doped).

num_inter is **int** type. The default is 6, and it is **optional**.

Example of *num_inter* setting:

```
# Default:
num_inter = 6

# The greater num_inter is, the more interstitial will be generated, which will increase␣
↪the amount of calculation.

# Example, please revise by the self.
num_inter = 10
```

3.4.13 3.4.13 inter_host_distance

inter_host_distance indicates the minimum distance between the interstitial atom and other atoms when interstitial defects are generated.

inter_host_distance is **float** type. The default is 1.6, and it is **optional**.

Example of *inter_host_distance* setting:

```
# Default:
inter_host_distance = 1.6

# The greater inter_host_distance is, the slower the interstitial defect is generated.
# Example, please revise by the self.
inter_host_distance = 1.4
```

Note: Please be cautious when using this parameter! The great inter_host_distance may cause interstitial atoms cannot find their position all the time because they are generated by random scattering, so the DASP-DEC cannot continue. Using the default value generally takes several minutes, and the generation time increases with the increase of this value. If there is no response for a long time (more than ten minutes), please reset this parameter and run DASP-DEC again.

3.4.14 3.4.14 inter_inter_distance

inter_inter_distance indicates the distance between two generated interstitial atoms.

inter_inter_distance is **float** type. The default is 0.1, and it is **optional**.

Example of *inter_inter_distance* setting:

```
# Default:
inter_inter_distance = 0.1

# The greater inter_inter_distance is, the slower the interstitial defect is generated.
# Example, please revise by the self.
inter_inter_distance = 0.2
```

Note: Please be cautious when using this parameter! The great inter_inter_distance may cause interstitial atoms cannot find their position all the time because they are generated by random scattering, so the DASP-DEC cannot continue. Using the default value generally takes several minutes, and the generation time increases with the increase of this value. If there is no response for a long time (more than ten minutes), please reset this parameter and run DASP-DEC again.

3.4.15 3.4.15 correction

correction indicates the finite supercell size corrections for charged defects. None means only electrostatic potential alignment without image charge correction. LZ means the electrostatic potential alignment and Lany-Zunger image charge correction are used. FNV means using FNV correction scheme (including electrostatic potential alignment).

correction is **str** type. The default is None, and it is **optional**.

Example of *correction* setting:

```
# Default:
correction = None

# LZ correction
correction = LZ/lz/Lz/lZ   # case insensitive

# FNV correction
correction = FNV/fnv/...   # case insensitive

# References: Phys. Rev. B 78, 235104 (2008), Phys. Rev. Lett. 102, 016402 (2009)
```

3.4.16 3.4.16 epsilon

epsilon indicates the static dielectric constant required to calculate the correction of charged defects.

epsilon is **float** type. No default, and it is **optional**. **(It is required when correction = LZ or FNV.)**

Example of *epsilon* setting:

```
# Default: no default
epsilon =

# Example, please revise by the self.
epsilon = 12.6
```

3.4.17 3.4.17 Eg_real

Eg_real indicates the experimental band gap used to calculate the exchange proportion AEXX for HSE calculation.

Eg_real is **int** type. No default, and it is **optional**.

Example of *Eg_real* setting:

```
# Default: no default
Eg_real =

# Valid only when level   1.
# Eg_real > 0, if there is no experimental band gap, it does not have to be set.
# 0.25 will be used for subsequent calculation.
```

(continues on next page)

(continued from previous page)

```
# Example, please revise by the self.
Eg_real = 5.6
```

3.4.18 3.4.18 distorted_defect

distorted_defect indicates the defect name of the distorted structure to be generated, which must be the same as the name of the defect folder (except for interstitial). Interstitial defects are slightly different, for example, N_i/random2 corresponds to N_i-2.

distorted_defect is **list** type. No default, and it is **optional**. If set, the distortion structure will be calculated.

Example of *distorted_defect* setting:

```
# Default: no default
distorted_defect =

# Example, please revise by the self.
distorted_defect = Ga_N2 V_N1 N_i-2
```

3.4.19 3.4.19 distorted_number

distorted_number indicates the number of distorted structures to be generated of each defect.

distorted_number is **int** type. The default is 10, and it is **optional**.

Example of *distorted_number* setting:

```
# Default:
distorted_number = 10

# Valid when the distorted_defect is set correctly.

# Example, please revise by the self.
distorted_number = 6
```

3.5 3.5 DDC parameters

3.5.1 3.5.1 ddc_temperature

ddc_temperature indicates the growth and working (measuring) temperature, two values have to be set.

ddc_temperature is **list** type. No default, and **required**.

Example of *ddc_temperature* setting:

```
# Default: no default
ddc_temperature =
```

(continues on next page)

(continued from previous page)

```
# Example, please revise by the self.
# The growth temperature is 1000 K, while the working (measuring) temperature is 300 K.
ddc_temperature = 1000 300
# ddc_temperature = 1300 330
```

3.5.2 3.5.2 ddc_mass

ddc_mass indicate the electron and hole effective masses, the geometric mean in the three directions x, y, and z of the carrier effective mass should be manually calculated before filling.

ddc_mass is **list** type. No default, and **required**.

Example of *ddc_mass* setting:

```
# Default: no default
ddc_mass =

# Example, please revise by the self.
# The electron effective mass is 0.1m_0, and the hole effective mass is 0.9m_0.
ddc_mass = 0.1 0.9
```

3.5.3 3.5.3 ddc_path

ddc_path indicates the path of chemical potential calculated by DDC. The serial number is consistent with that in `dasp.in` , and two values must be set.

ddc_path is **list** type. The default is **1 2** , which means the defect concentration on the path from the first chemical potential (p1) to the second chemical potential (p2) setting in dasp.in will be calculated, and it is **optional**.

Example of *ddc_path* setting:

```
# Default:
ddc_path = 1 2

# Example, please revise by the self.
# It indicates the changes in defect concentration on the path from the fourth chemical␣
↪potential to the second chemical potential that setting in dasp.in.
ddc_path = 4 2
```

3.6 3.6 CDC parameters

The CDC module will read the four parameters of *level* , *epsilon* , *vasp_path_cdc* , and *ddc_mass* mentioned above, and the following seven parameters.

3.6.1 3.6.1 cdc_defect

cdc_defect specifies the defects to be calculated in CDC.

cdc_defect is **str** type. No default, and it is **required**.

Example of *cdc_defect* setting:

```
# Default: no default
cdc_defect =

# Example, please revise by the self.
# Calculating the property of Cu_Zn1.
cdc_defect = Cu_Zn1
```

Note: The name of the specified defect must be consistent with the directory name of the defect in the dec directory.

Note: Only the properties of the specified defect in the DEC directory with "initialstructure" can be calculated.

3.6.2 3.6.2 cdc_job

cdc_job determines which calculation needs to do in CDC. radiative_rate will calculate the radiative capture coefficient, and pl calculates the photoluminescence spectra induced by defects.

cdc_job is **str** type. No default, and it is **required**.

Example of *cdc_job* setting:

```
# Default: no default
cdc_job =

# Calculating the radiative capture coefficient.
cdc_job = radiative_rate

# Calculating the photoluminescence spectra.
cdc_job = pl
```

3.6.3 3.6.3 cdc_temperature

cdc_temperature specifies the temperature of material when calculating the defect properties in CDC.

cdc_temperature is **float** type. No default, and it is **required**.

Example of *cdc_temperature* setting:

```
# Default: no default
cdc_temperature =
```

(continues on next page)

(continued from previous page)

```
# Calculating the properties when the material is at 300 K.
cdc_temperature = 300
```

3.6.4 3.6.4 cdc_charge

cdc_charge represents the amount of charge of the defect before and after carrier transition.

cdc_charge is **list** type. No default, and it is **required**.

Example of *cdc_charge* setting:

```
# Default: no default
cdc_charge =

# It indicates that a hole transfers to the defect state, the initial is a neutral state␣
↪with 0 charges, while the final state is a +1 state with a +q charge.
cdc_charge = 0 1

# It indicates that an electron transfers to the defect state, the initial is a -1 state␣
↪with a -q charge, while the final state is a -2 state with a -2q charge.
cdc_charge = -1 -2
```

3.6.5 3.6.5 cdc_band

cdc_band sets which band the carrier is located before and after the transition. If it is a hole transition, the hole is located at the VBM and defect state before and after transition respectively. If it is an electron transition, the electron is located at the CBM and defect state before and after transition respectively.

cdc_band is **list** type. No default, and it is **required**.

Example of *cdc_band* setting:

```
# Default: no default
cdc_band =

# This is a hole transition, before and after the transition the hole is located at the␣
↪864 band which is the VBM, and at the 865 band which is the defect state respectively.
cdc_band = 864 865

# This is an electron transition, before and after the transition the electron is␣
↪located at the 866 band which is the CBM, and at the 865 band which is the defect␣
↪state respectively.
cdc_band = 866 865
```

Note: If cdc_band corresponds to electron transition, the cdc_charge should also be. The same for hole transition.

3.6.6 3.6.6 spin_channel

spin_channel indicates the spin of the carrier. 1 corresponds to spin up or no spin, and 2 corresponds to spin down.

spin_channel is **int** type. No default, and it is **required**.

Example of *spin_channel* setting:

```
# Default: no default
spin_channel =

# 1 corresponds to spin up or no spin.
spin_channel = 1

# 2 corresponds to spin down.
spin_channel = 2
```

3.6.7 3.6.7 refractive_index

refractive_index the refractive index of the material.

refractive_index is **float** type. No default, and it is **required**.

Example of *refractive_index* setting:

```
# Default: no default
refractive_index =

# The refractive index of the material is 2.38.
refractive_index = 2.38
```

4

4 Output

4.1 4.1 Output of PREPARE module

All the outputs of PREPARE module are in the dec directory, and the following is the directory tree:

```
dec
   1prepare.out
   dasp.in
   madelung
           INCAR
           KPOINTS
           ...
   AEXX
       0.25
           static
               INCAR
               KPOINTS
               ...
       0.3
           static
               ...
   relax
       ...
```

4.1.1 4.1.1 INCAR

It is the input file for VASP calculation. This module will output two kinds of INCAR files including `INCAR-relax` (INCAR for structure relaxation) and `INCAR-static` (INCAR for static calculation).

4.1.2 4.1.2 POSCAR

It is the input file for VASP calculation. This module will output files including `POSCAR` (provided by the user), `POSCAR_refined` (POSCAR for the refined cell), `POSCAR_nearlycube` (POSCAR of nearly cubic supercell), and `POSCAR_final` (POSCAR of the relaxed supercell).

4.1.3 4.1.3 KPOINTS

It is the input file for VASP calculation. This module will output files as `KPOINTS` . Note: KPOINTS generated by DASP is always Gamma only. Do not replace KPOINTS with multiple k points.

4.1.4 4.1.4 POTCAR

It is the input file for VASP calculation. This module will output files including `POTCAR` (POTCAR for intrinsic defect calculation) and `POTCAR_X` (POTCAR for X dopant).

4.1.5 4.1.5 Submission script

It is the script used to submit job for DFT calculation.

4.1.6 4.1.6 Directory of Madelung constant calculation

The madelung directory would be created in the dec directory by PREPARE to calculate Madelung constant, and output it to the parameter *madelung* in `dasp.in` .

4.1.7 4.1.7 Directory of HSE exchange parameter calculation

If the *level=2 or 3* and *Eg_real* is set, the AEXX directory will be created in the dec directory to calculate exchange proportion and output to the corresponding setting in INCAR.

4.1.8 4.1.8 Directory of Supercell optimization

No matter how *level* is set, the supercell optimization is always carried out under the relax directory.

4.1.9 4.1.9 1prepare.out

All details of running, and results and errors on this module are written into the file `1prepare.out` . The user can query the execution status in this file.

4.2 4.2 Output of TSC module

All the outputs of TSC module are in the tsc directory, and the following is the directory tree of TSC:

```
tsc
   2tsc.out
   dasp.in
   materials_info.yaml
   materials_info_recalc.yaml
   pure_phase_energy.yaml
   key_phases_info_recalc.yaml
   host compound
       relaxation1
           INCAR ...
       relaxation2
           INCAR ...
       static
           INCAR ...
       static_recalc
           INCAR ...
   secondary compound 1
       static_recalc
           INCAR ...
   secondary compound 2
       static_recalc
           INCAR ...
```

4.2.1 4.2.1 materials_info.yaml

`materials_info.yaml` is the output file when TSC performs the first calculation and analysis. The file contains the thermodynamic stability results of the host compound, including the following information:

- `FE` : the formation energy of the host compound (eV/per atom).
- `TE` : the total energy of the host compound (eV/per atom).
- `TS` : the thermodynamic stability of the host compound (True or False).
- `chemical_potentials` : the chemical potential range of each element of the host compound.
  - < element name>
  - < the maximum value of the chemical potential of the element (eV) >
  - < the minimum value of the chemical potential of the element (eV) >
- `decomp` : the most probable path of decomposition of the host compound.
  - < name of each decomposition >

- `direct` : the band gap of the host compound is direct or indirect.
- `e_above_hull` : the energy difference between the host compound and the equilibrium phase with the same composition, namely Energy above hull (eV/per atom). It is positive when the material is unstable, otherwise, it is negative.
- `gap` : the band gap of the host compound (eV).
- `hull_points` : the chemical potential of the elements at the boundary of the stable region of the host compound.
  - < element name >
  - < the chemical potential of each element at p1 (eV) >
  - < the chemical potential of each element at p2 (eV) >
  - ...
- `key_phases` : the critical competing compounds that limit the stability.
  - < the name of the critical competing compounds >
- `XX_doped_key_phases` : the critical competing compounds used to calculate the chemical potential of dopants X.
  - < the name of the critical competing compounds >
- `key_phases_2d` : the critical competing compounds that affect the stability of the host compounds in the two-dimensional stable region phase diagram.
  - < the name of the critical competing compounds >
- `secondary_phases` : all the considered competing secondary compounds in the analysis of the stability of the host compound.
  - < the name of the secondary compounds >
  - < the total energy of the secondary compounds (eV/ per atom) >
  - < the formation energy of the secondary compounds (eV/ per atom) >

4.2.2 4.2.2 materials_info_recalc.yaml

`materials_info_recalc.yaml` is the output file when TSC performs the second calculation and analysis. The file structure is similar to that of `materials_info.yaml` .

4.2.3 4.2.3 pure_phase_energy.yaml

`pure_phase_energy.yaml` is the intermediate output file when TSC performs the first calculation and analysis. It recorded the information about the pure phases extracted from the MP database or provided by this module during the first calculation and analysis.

- < the name of the pure phase>
  - `TE` : total energy of the pure phase (eV/per atom).
  - `cif` : the structure of the pure phase.
  - `id` : the id of the pure phase in the MP database.
  - `space group` : the space group information of the pure phase.

For the pure phases H2, O2, N2, F2, Cl2 and Br2, the `id = 0` , `space group = null` , and `cif = null` , due to the structure and energy provided by this module adopted.

4.2.4 4.2.4 key_phases_info_recalc.yaml

`key_phases_info_recalc.yaml` is the intermediate output file when TSC performs the second calculation and analysis. It recorded the information about the key hetero-phases involved in the second calculation and analysis.

- < the name of the key hetero-phases>
  - `TE` : total energy of the key hetero-phases (eV/ per atom).
  - `direct` : total energy of the key hetero-phases (eV/ per atom).
  - `gap` : the band gap of the key hetero-phases (eV).

4.2.5 4.2.5 stable_2d.out

`stable_2d.out` is the output file when TSC performs the first stage analysis, which is located in the directory tsc/2d-figures/. It recorded the chemical potentials of the elements at the boundary of the two dimensional stable region. Meanwhile, for ternary or multinary compounds, the stable region picture with png format will output, corresponding to the analysis of the first stage. The specific content can be seen in the case description, and the format is as follows.

```
<element>            <element>            <element>
-------------------  -------------------  -------------------
<chemical potential> <chemical potential> <chemical potential>
<chemical potential> <chemical potential> <chemical potential>
<chemical potential> <chemical potential> <chemical potential>
<chemical potential> <chemical potential> <chemical potential>
```

4.2.6 4.2.6 stable_recalc_2d.out

`stable_recalc_2d.out` is the output file when TSC performs the second stage analysis, which is located in the directory tsc/2d-figures/. It recorded the chemical potentials of the elements at the boundary of the two dimensional stable region. The specific content can be seen in the case description, the format is the same as `stable_2d.out` . Meanwhile, for ternary or multinary compounds, the stable region picture with png format will be output, corresponding to the analysis of the second stage.

4.2.7 4.2.7 stable.out

`stable.out` is the output file when TSC performs the first stage analysis, which is located in the directory tsc/3d-figures/. It recorded the chemical potentials of the elements at the boundary of the three dimensional stable region. The specific content can be seen in the case description, and the format is as follows.

```
<element>            <element>            <element>
-------------------  -------------------  -------------------
<chemical potential> <chemical potential> <chemical potential>
```

(continues on next page)

(continued from previous page)

```
<chemical potential> <chemical potential> <chemical potential>
<chemical potential> <chemical potential> <chemical potential>
<chemical potential> <chemical potential> <chemical potential>
```

4.2.8 4.2.8 stable_recalc.out

`stable_recalc_2d.out` is the output file when TSC performs the second stage analysis, which is located in the directory tsc/3d-figures/. It recorded the chemical potentials of the elements at the boundary of the three dimensional stable region. The specific content can be seen in the case description, the format is the same as `stable.out` .

4.2.9 4.2.9 ori_data_MP

The detailed output files of the first stage analysis of TSC module are retained in the directory `ori_data_MP` , including the subdirectory `all_cif\` which retained all hetero-phase structure files, and the data `DataRecord_XX.pkl` extracted from the MP database and corresponding subdirectory `XX\` . Generally, The name of the subdirectory `XX\` consists of the name of the host compound (and the doped element), which retains the hetero-phases structure and data for the corresponding case.

4.2.10 4.2.10 2tsc.out

`2tsc.out` is the log file during the execution of TSC module, including the progress of calculation and analysis and possible error messages. The specific content can be seen in the case description.

4.3 4.3 Output of DEC module

All the outputs of DEC module are in the dec directory, and the following is the directory tree of DEC:

```
dec
   3dec.out
   dasp.in
   Formation_Energy_Intrinsic_Defect
       p1.dat
       p2.dat
       p1.png
       p2.png
   Transition_Level_Intrinsic_Defect
       tl.dat
       tl.png
   defect 1
   defect 2
   defect 3
...
```

4.3.1 4.3.1 Formation_Energy_(Intrinsic_Defect)

This directory contains the pictures of the formation energy of intrinsic or dopants (subject to the folder name) as a function of Fermi level, including raw data with .dat format, as `p1.dat, p2.dat, ...` , and pictures with png format, as `p1.png, p2.png, ...` , The user can open and compile the .dat file using origin, and draw by self.

4.3.2 4.3.2 Transition_Level_(Intrinsic_Defect)

This directory contains the pictures of the transition energy level of intrinsic or dopants (subject to the folder name), including raw data with .dat format like `tl.dat` , and pictures with png format like `tl.png` . The user can open and compile the .dat file using origin, and draw by self.

4.3.3 4.3.3 3dec.out

All details of running, and results and errors on this module are written into the file `3dec.out` . The user can query the execution status in this file. The specific content can be seen in the case description.

4.4 4.4 Output of DDC module

All the outputs of DDC module are in the ddc directory, and the following is the directory tree of DDC:

```
ddc
   4ddc.out
   DefectParams.txt
   Fermi.dat
   Carrier.dat
   Defect_charge.dat
```

4.4.1 4.4.1 DefectParams.txt

`DefectParams.txt` contains the required input parameters for DDC calculation, including **defect name**, **Nsites**, **degeneracy factor**, **transition energy level**, and **formation energy of neutral defect**. Taking ZnGeP2 as example, after run this module `DefectParams.txt` :

```
1000 300
0.360000 0.190000
2.067143
P_Zn1 1.245078e+22 2 1.0959 1 0.7854 2 0.602 1 0.3478 2 1.3758 1 1.8193 2 1.8431 1 2.
↪0373 2 3.993075 2.591475
Zn_P1 2.490156e+22 2 0.9803 1 0.4186 2 0.1979 1 x x 1.3004 1 1.6611 2 1.8974 1 2.0551 2␣
↪2.013776 3.415376
Ge_P1 2.490156e+22 2 0.017 1 x x x x x x 0.6089 1 1.4605 2 1.7595 1 x x 1.214661 1.681861
P_Ge1 1.245078e+22 2 1.5928 1 0.6776 2 0.3533 1 x x 1.8788 1 2.0787 2 x x x x 2.242467 1.
↪775267
...
...
```

The script, dissected

```
# The growth temperature is 1000 K and the working (measuring) temperature is 300 K, as␣
↪read from file dasp.in.
1000 300
```

```
# The electron effective mass is 0.36m_0 and that of hole is 0.19m_0, as read from file␣
↪dasp.in.
0.360000 0.190000
```

```
# Tha band gap of supercell as read from dec/Intrinsic_Defect/host/EIGENVAL or dec/
↪Doping_XX/host/EIGENVAL.
2.067143
```

```
# From left to right: defect name, Nsites, degeneracy factor for q=0, transition energy␣
↪level (0/+), degeneracy factor for q=+1, transition energy level (0/2+), degeneracy␣
↪factor for q=+2, transition energy level (0/3+), degeneracy factor for q=+3,␣
↪transition energy level (0/4+), degeneracy factor for q=+4, transition energy level (0/
↪-), degeneracy factor for q=-1, transition energy level (0/2-), degeneracy factor for␣
↪q=-2, transition energy level (0/3-), degeneracy factor for q=-3, transition energy␣
↪level (0/4-), degeneracy factor for q=-4, formation energy 1, formation energy 2.
# Formation energy 1 and formation energy 2 correspond to the neutral defect formation␣
↪energy when the chemical potential (in dasp.in) at p1 and p2, respectively.
# From the fourth row, there are 21 strings in every row, all the data is from the␣
↪output of defect calculation in dec directory.
P_Zn1 1.245078e+22 2 1.0959 1 0.7854 2 0.602 1 0.3478 2 1.3758 1 1.8193 2 1.8431 1 2.
↪0373 2 3.993075 2.591475
```

```
# In the same order as the previous row, there are 21 strings.
# If one charge state does not exist, its transition energy level and degeneracy factor␣
↪will be denoted by x. For example, there has no q=+4 state, so the (0/4+) level and␣
↪degeneracy factor for q=+4 is x.
Zn_P1 2.490156e+22 2 0.9803 1 0.4186 2 0.1979 1 x x 1.3004 1 1.6611 2 1.8974 1 2.0551 2␣
↪2.013776 3.415376
```

```
# Similarly, there has no q=+2, +3, +4, and -4 state, so their levels and degeneracy␣
↪factors are x.
Ge_P1 2.490156e+22 2 0.017 1 x x x x x x 0.6089 1 1.4605 2 1.7595 1 x x 1.214661 1.681861
```

```
# Similarly, there has no q=+4, -3, and -4 state, so their levels and degeneracy factors␣
↪are x.
P_Ge1 1.245078e+22 2 1.5928 1 0.6776 2 0.3533 1 x x 1.8788 1 2.0787 2 x x x x 2.242467 1.
↪775267
```

Note: The DDC module can be used independently of DASP-TSC and DASP-DEC, namely, the user only needs to prepare DefectParams.txt as the format described in this section, and DASP can treat it as an input and perform calculations. The degeneracy factor of the defect does not affect the order of magnitude of the concentration, so it can be set to 1 if uncertain.

4.4.2 4.4.2 Fermi.dat

The output of `Fermi.dat` has two columns. The first column shows the linear change of chemical potential from p1 to p2 (chemical potential in dasp. in). And the second column is the Fermi level corresponding to the chemical potential. EF=0 indicates that the Fermi level is located in VBM.

4.4.3 4.4.3 Carrier.dat

The output of `Carrier.dat` has three columns. The first column shows the linear change of chemical potential from p1 to p2 (chemical potential in dasp. in). The second and third columns are the electron and hole (free) carrier densities corresponding to the chemical potential respectively.

4.4.4 4.4.4 Defect_charge.dat

The output of `Defect_charge.dat` has multiple columns. The first column shows the linear change of chemical potential from p1 to p2 (chemical potential in dasp. in). From the second column, every column is the concentration of defects on different charge states at the corresponding chemical potential.

4.4.5 4.4.5 4ddc.out

All details of running, and results and errors on DDC module are written into the file `4ddc.out` . The user can query the execution status in this file. The specific content can be seen in the case description.

4.5 4.5 Output of CDC module

All the outputs of CDC module are in the cdc directory, and the following is the directory tree of CDC:

```
cdc
   5cdc.out
   dasp.in
   pl_calc.list
   POSCAR
   specify defect
          Radiate_calc
              specify the initial and final state
                  final_state
                  initial_state
                  intermediate_state
                  pl.out
                  rad_rate.out
                  lineshape.dat
                  lineshape.png
                  ccdiagram.png
          ...
```

4.5.1 4.5.1 5cdc.out

All details of running, and results and errors on CDC module are written into the file `5cdc.out` . The user can query the execution status in this file. The specific content can be seen in the case description.

4.5.2 4.5.2 lineshape.dat & lineshape.png

The file `lineshape.dat` is the raw data of the output after PL spectrum calculation, while `lineshape.png` is the image of PL spectrum on the basis of `lineshape.dat` . The specific content can be seen in the case description.

4.5.3 4.5.3 ccdiagram.png

The image `ccdiagram.png` is the configuration coordinate diagram when the defects are in initial and final states. The specific content can be seen in the case description.

4.5.4 4.5.4 pl.out

The file `pl.out` is a backup of `5cdc.out` after the PL spectrum calculation is completed.

4.5.5 4.5.5 rad_rate.out

The file `rad_rate.out` is a backup of `5cdc.out` after the radiative capture coefficient calculation is completed.

5

5 Examples

5.1 5.1 The calculations of intrinsic defects in CdTe

CdTe is a binary photovoltaic semiconductor with a direct band gap of 1.45 eV. Based on the Shockley-Queisser theory, its theoretical maximum efficiency can be nearly 30%. However, photogenerated carriers can be recombined through defect levels of the point defects, especially the recombination centers, which reduces the carrier lifetime. Meanwhile, the carriers induced by the ionized defects can also change the Fermi level, and change the conductivity of CdTe. Therefore, it is necessary to calculate the defect properties of CdTe under different growth conditions in detail.
The following is an example of using DASP to calculate intrinsic defects in CdTe:

5.1.1 5.1.1 PREPARE --- Prepares for calculation

5.1.1.1 POSCAR and dasp.in

Find the POSCAR of the primitive cell of CdTe from the `Materials Project` database, as follows:

```
Cd1 Te1
1.0
4.6874446869          0.0000000000          0.0000000000
2.3437223434          4.0594461777          0.0000000000
2.3437223434          1.3531487259          3.8272825601
Cd   Te
1    1
Direct
0.000000000          0.000000000          0.000000000
0.750000000          0.750000000          0.750000000
```

Using the crystal visualization software, the structure can be seen in Fig.1:

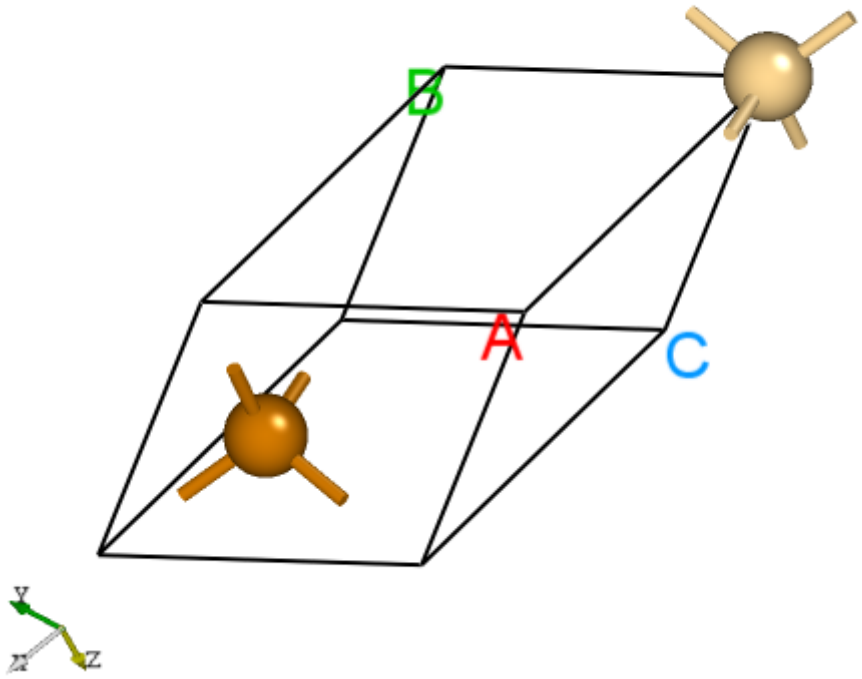


Fig.1 The primitive cell of CdTe.

Next, use VASP to optimize its lattice parameters or modify the lattice to match the experimental measurements. Users need to do it manually.

Write the required parameters in `dasp.in` :

```
############### Job Scheduling ###############
cluster = SLURM     # (job scheduling system)
node_number = 2          # (number of node)
core_per_node = 52       # (core per node)
queue = batch            # (name of queue/partition)
max_time = 24:00:00         # (maximum time for a single DFT calculation)
vasp_path_dec = /opt/vasp.5.4.4/bin/vasp_gam   #  (path of VASP)
vasp_path_tsc = /opt/vasp.5.4.4/bin/vasp_std
job_name = submit_job    # (name of script)
potcar_path = /opt/POT/potpaw_PBE    # (path of pseudopotentials)
max_job = 5
```

(continues on next page)

(continued from previous page)

```
############## TSC Module ##############
database_api = ******************** # (str-list type)

############## DEC Module ##############
level = 2   # (level=1: PBE+PBE; level=2: PBE+HSE; level=3: HSE+HSE)
min_atom = 190
max_atom = 240
intrinsic = T   # (default: T)
correction = FNV   # (default: none)
epsilon = 10.3
Eg_real = 1.45   # (experimental band gap)

############## DDC Module ##############
ddc_temperature = 1000 300
ddc_mass = 0.09 0.84
```

Next, all the parameters listed in `dasp.in` will be described:

```
cluster = SLURM
# The system of the used cluster is SLURM.
```

```
node_number = 2
# 2 nodes are used for each calculation.
```

```
core_per_node = 52
# 52 cores are used for each node, so 2*52=104 cores are used in total for each␣
↪calculation.
```

```
queue = batch
# The queue named "batch" is used to carry out calculations. Therefore, users need to␣
↪make sure the queue name, nodes, and cores of clusters before configuring dasp.in.
```

```
max_time = 24:00:00        # (maximum time for a single DFT calculation)
# The maximum time allowed is 24 hours for a single DFT calculation and can be set␣
↪arbitrarily.
```

```
vasp_path_dec = /opt/vasp.5.4.4/bin/vasp_gam   #  (path of VASP)
vasp_path_tsc = /opt/vasp.5.4.4/bin/vasp_std
# The VASP_std version is used for TSC calculations, and the VASP_gam version is used␣
↪for DEC calculations.
```

```
job_name = submit_job    # (name of script)
# The submission script, named "submit_job" and can be set arbitrarily.
```

```
potcar_path = /opt/POT/potpaw_PBE    # (path of pseudopotentials)
# path of pseudopotentials
```

```
max_job = 5
# the allowed maximum number of jobs at the same time
```

```
database_api = ******************** # (str-list type)
# using to visit the Materials Project database
```

```
level = 2   # (level=1: PBE+PBE; level=2: PBE+HSE; level=3: HSE+HSE)
# using GGA-PBE for structural relaxation and HSE to calculate the total energy
```

```
min_atom = 190
max_atom = 240
# The number of atoms within the generated supercell that we want is between 190 and 240,
↪ and as far as possible to make a=b=c and a b c.
```

```
intrinsic = T   # (default: T)
# Generate intrinsic defects, V_Cd, V_Te, Cd_Te, Te_Cd, Cd_i, and Te_i.
```

```
correction = FNV   # (default: none)
# The corrections for charged defect adopt FNV correction.
```

```
epsilon = 10.3
# The dielectric constant of CdTe is 10.3.
```

```
Eg_real = 1.45   # (experimental band gap)
# The experimental band gap of CdTe is about 1.45 eV, DASP will adjust AEXX in INCAR to␣
↪make the band gap of the supercell without defect equal to 1.45 eV.
```

```
ddc_temperature = 1000 300
# the growth temperature set to 1000 K and the working temperature set to 300 K.
```

```
ddc_mass = 0.09 0.84
# electron effective mass set to 0.09 and hole effective mass set to 0.84.
```

5.1.1.2 Use DASP to generate the required input files

Create a new directory CdTe, then prepare the files, `POSCAR` and `dasp.in` , mentioned above in the directory ./CdTe/. Next, execute `dasp 1` to start PREPARE module and no additional operation is needed thereafter. DASP will output file `1prepare.out` to record the running log of the module.

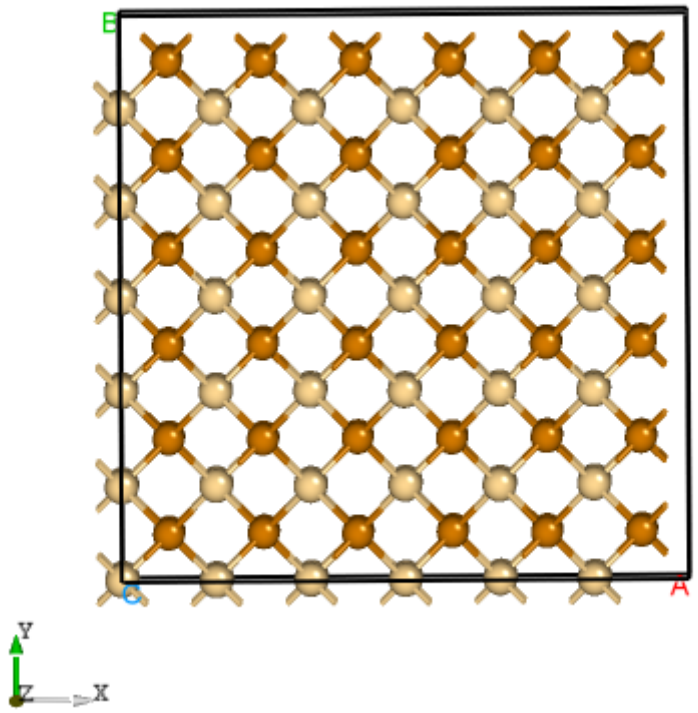
5.1.1.3 Workflow of PREPARE module

Generate supercell:

Firstly, the program will automatically find the optimal supercell (as far as possible to make a=b=c and a b c) based on the parameters min_atom=190 and max_atom=240 and output POSCAR for the supercell. The following are the structure messages of supercell `POSCAR_nearlycube` expanded by CdTe primitive cell.

```
Cubic_cell
1.0
19.8871435472 0.0000000000 0.0000000000
0.0000000000 19.8871435472 0.0000000000
0.0000000000 0.0000000000 19.8871435472
Cd Te
108 108
Direct
0.0000000000 0.0000000000 0.0000000000
0.8333333333 1.0000000000 0.1666666666
0.8333333333 0.1666666666 1.0000000000
0.6666666666 0.1666666666 0.1666666666
...
```

Through the visualization software, we can see that the included angle of the axis of the given CdTe cell is small, but the supercell generated by DASP is vertical on three sides.



The supercell of CdTe.

Madelung constant calculation:

Secondly, according to the generated supercell, the program will execute Madelung constant calculation which describes the Coulomb interaction between charged defect and periodic image charge. (use for Lany-Zunger correction)

After finishing the above two steps calculation, the output of `1prepare.out` is as follows:

```
############ Prepare Files module start ############

Read the structure file POSCAR you provided
Get the refined cell POSCAR_refined from POSCAR
Generate the nearlycube cell POSCAR_nearlycube from POSCAR
Generate job script through dasp.in parameters
Generate single-point KPOINTS
Generate pseudopotential file POTCAR through potcar_dir you set
Generate commonly used vasp input file INCAR
Start the madelung constant calculation
Generate the madelung calculation directory
Generate madelung calculation POSCAR
Generate madelung calculation POTCAR
Generate madelung calculation INCAR
Generate madelung calculation KPOINTS
Generate madelung calculation job script
Job 103.host5 submitted: /home/test/CdTe/dec/madelung/static
Succeed job 103.host5: /home/test/CdTe/dec/madelung/static
The madelung constant calculation completed
The madelung constant = 2.837
```

HSE exchange proportion calculation:

According to the generated supercell, the program will perform HSE static calculations with AEXX=0.25 and AEXX=0.3 respectively to determine the value of AEXX which can make the obtained band gap match *Eg_real = 1.45* based on the slope. Therefore, after those calculations are completed, the contents in the directory CdTe/dec/AEXX/ are as follows:

```
cd ./dec/AEXX
ls
0.25  0.25880073638027207  0.3  AEXX.list
```

It indicates that the band gap of the CdTe supercell is 1.45 eV when AEXX=0.26 (two decimal places), and write this parameter into INCAR. Meanwhile, the log can be seen from `1prepare.out` as follows:

```
Start the HSE parameter AEXX calculation
Job 107.host5 submitted: /home/test/CdTe/dec/AEXX/0.25/static
Job 108.host5 submitted: /home/test/CdTe/dec/AEXX/0.3/static
Succeed job 107.host5: /home/test/CdTe/dec/AEXX/0.25/static
Succeed job 108.host5: /home/test/CdTe/dec/AEXX/0.3/static
Job 108.host5 submitted: /home/test/CdTe/dec/AEXX/0.25880073638027207/static
Succeed job 108.host5: /home/test/CdTe/dec/AEXX/0.25880073638027207/static
The HSE parameter AEXX calculation completed
The HSE parameter AEXX = 0.26
level = 2: Generate PBE relax vasp input file INCAR-relax
level = 2: Generate HSE static vasp input file INCAR-static
```

Optimize the ionic position of the host supercell:

The last step in PREPARE module is to optimize the ionic position of the host supercell according to `level=2` (PBE relax). The optimized file is `POSCAR_final` in the directory CdTe/dec/relax. At the same time, the sign of the end of DASP operation can be seen in `1prepare.out` , and it also tells us that we need to do the TSC module calculation in the next step.

```
Start the POSCAR_nearlycube relax calculation
Generate the POSCAR_nearlycube relax directory
Job 109.host5 submitted: /home/test/CdTe/dec/relax
Succeed job 109.host5: /home/test/CdTe/dec/relax
The POSCAR_nearlycube relax calculation completed
Get the final structure POSCAR_final

############ Prepare Files module end ############

PREPARE module finished, please run DASP-TSC next
```

5.1.2 5.1.2 TSC -- thermodynamic stability and chemical potential calculations

5.1.2.1 Run TSC module

The directory CdTe/dec will be created when using the command `dasp 1` to execute PREPARE module, and generate file `1prepare.out` in this directory. After finishing the program, there has the corresponding completion flag in `1prepare.out` . Then, enter the directory CdTe/dec and confirm that the parameters in `INCAR-relax` and `INCAR-static` are feasible. (Users can modify INCAR, and DASP will make subsequent calculations based on the INCAR in this directory.)

Once confirm that the PREPARE module is finished, return to the directory CdTe and use the command `dasp 2` to execute the TSC module. Similarly, the TSC module will create a directory named `tsc` under the directory CdTe, which contains the output of the TSC program, including every calculation directory and the running log file `2tsc.out` . No additional operation is required while waiting for the program to complete.

5.1.2.2 Workflow of TSC module

The total energy calculation of the host structure (the parameters are consistent with MP database):

TSC module will use the same input parameters (INCAR, KPOINTS, POTCAR) with the `Materials Project` database to perform structural relaxation and static calculation on the primitive cells given by the user. Therefore, the calculated total energy is comparable to that of the MP database. This step is to obtain the **key hetero-phases** that limit the stability of CdTe. In the directory, we can see:

```
cd tsc
cd CdTe/
ls
relaxation1  relaxation2  static
```

The running log also can be seen from the CdTe/tsc/2tsc.out, that is, the steps such as generating input files, relaxation1, relaxation2, static and data extraction.

The judgement of key hetero-phases compounds:

The TSC module will search for all the secondary compounds that compete with CdTe in the MP database. And compare the total energy of CdTe calculated in the previous step with that of the hetero-phases extracted from the database to confirm CdTe is **thermodynamically stable**. Subsequently, the program will automatically download the key hetero-phases compounds that can limit the thermodynamic stability of CdTe. Only Cd and Te are considered in this case. The relevant information can be seen in `2tsc.out`:

```
...
analysing the thermodynamic stability of CdTe.
key phases of CdTe are: Cd Te .
file key_phases_info_recalc.yaml generated.
analysing of CdTe is done.
...
```

The total energy calculation of the host and hetero-phase compounds:

After the key hetero-phase compounds are confirmed, TSC will calculate the total energy of CdTe, Cd, and Te by using the parameter (AEXX) obtained from PREPARE module. `2tsc.out` is as follows:

```
...
Job 112.host5 submitted: /home/test/CdTe/tsc/CdTe/static_recalc
Job 113.host5 submitted: /home/test/CdTe/tsc/Cd/static_recalc
Job 114.host5 submitted: /home/test/CdTe/tsc/Te/static_recalc
Succeed job  112.host5: /home/test/CdTe/tsc/CdTe/static_recalc
Succeed job  113.host5: /home/test/CdTe/tsc/Cd/static_recalc
Succeed job  114.host5: /home/test/CdTe/tsc/Te/static_recalc
...
```

The chemical potential calculation:

Calculating the formation energy and stable chemical potential region of CdTe based on the calculated total energy. As CdTe is a binary compound, TSC module will give the endpoint of two chemical potentials, i.e. Cd-rich and Te-rich, and write them into `dasp.in` :

```
# The orders are consistent with the order of elements in POSCAR, i.e. the first column␣
↪is Cd and the second column is Te.
E_pure = -1.7736 -4.6974
p1 = 0.0 -1.1854
p2 = -1.1854 0.0
```

The output after the program is completed can be seen in `2tsc.out` :

```
dir '2d-figures','3d-figures','ori_data_MP' ready.  try to read file: 'calc_list.yaml'.
analysing the thermodynamic stability of CdTe.
key phases of CdTe are: Cd Te .
analysing of CdTe is done.

DASP-TSC finished
```

For the ternary and multinary compounds, TSC module will output the image of the stable region and the chemical potential at the endpoint of the stable region.

5.1.3 5.1.3 DEC -- the calculations of defect formation energy and transition energy level

5.1.3.1 Run DEC module

The directory CdTe/tsc will be created when using the command `dasp 2` to execute TSC module, and generate file `2tsc.out` in this directory. After finishing the TSC module, there has the corresponding completion flag in `2tsc.out` . Then, open the file `dasp.in` under the directory CdTe/dasp.in to confirm the chemical potential already has been written.

Once confirm that the TSC module is finished, return to the directory CdTe and use the command `dasp 3` to execute the DEC module. DEC will output relevant files in the generated directory dec in the first step, including the defect structures, directories, and the log file `3dec.out` . No additional operation is required while waiting for the program to complete.

5.1.3.2 Workflow of DEC module

Generate defect structure:

Based on the parameter intrinsic = T in `dasp.in` , DEC will generate the intrinsic defects for CdTe, i.e. create the calculation directory CdTe/dec/Intrinsic_Defect, in which the structures and directories of vacancies V_Cd and V_Te, antisite defects Cd_Te and Te_Cd, as well as interstitial defects Cd_i and Te_i are included. According to the crystal symmetry analysis, there has no inequivalent site for Cd and Te in CdTe lattice, thus only one configuration needs to be generated for each kind of defect.

```
cd dec/Intrinsic_Defect/
ls
Cd_i  Cd_Te1  host  Intrinsic_Defect.list  Te_Cd1  Te_i  V_Cd1  V_Te1
```

At the same time, the output of DEC module can be seen in `3dec.out` as follows:

```
############ Neutral Defect module start ############
Make intrinsic defect directory Intrinsic_Defect
Generate host directory in Intrinsic_Defect
Start generating neutral vacancy defect
Generate neutral defect at: V_Cd1/initial_structure/q0
Generate neutral defect at: V_Te1/initial_structure/q0
Neutral vacancy defect generation completed
Start generating neutral intrinsic antisite defect
Generate neutral defect at: Te_Cd1/initial_structure/q0
Generate neutral defect at: Cd_Te1/initial_structure/q0
Neutral intrinsic antisite defect generation completed
Start generating neutral intrinsic interstitial defect
Generate neutral defect at: Cd_i/random1/initial_structure/q0
Generate neutral defect at: Cd_i/random2/initial_structure/q0
Generate neutral defect at: Cd_i/random3/initial_structure/q0
Generate neutral defect at: Cd_i/random4/initial_structure/q0
Generate neutral defect at: Cd_i/random5/initial_structure/q0
Generate neutral defect at: Cd_i/random6/initial_structure/q0
Generate neutral defect at: Te_i/random1/initial_structure/q0
Generate neutral defect at: Te_i/random2/initial_structure/q0
Generate neutral defect at: Te_i/random3/initial_structure/q0
Generate neutral defect at: Te_i/random4/initial_structure/q0
Generate neutral defect at: Te_i/random5/initial_structure/q0
Generate neutral defect at: Te_i/random6/initial_structure/q0
Neutral intrinsic interstitial defect generation completed

############ Neutral Defect module end ############
```

It can be seen that the DEC module only creates the directories for all the neutral defects at this time.

Submit jobs for all defects with q=0:

After the structures and directories of neutral defects are generated, DEC module will call VASP to perform structural relaxation with PBE and total energy calculation with HSE (corresponds to the parameter level = 2 in `dasp.in`), this step may need a long time. Users can check the file `3dec.out` at any time. The messages in `3dec.out` are as follows:

```
Job 245.host5 submitted: /home/test/CdTe/dec/Intrinsic_Defect/Cd_Te1/initial_structure/q0
Job 246.host5 submitted: /home/test/CdTe/dec/Intrinsic_Defect/Te_Cd1/initial_structure/q0
Job 247.host5 submitted: /home/test/CdTe/dec/Intrinsic_Defect/V_Te1/initial_structure/q0
Job 248.host5 submitted: /home/test/CdTe/dec/Intrinsic_Defect/Te_i/random3/initial_
↪structure/q0
Job 249.host5 submitted: /home/test/CdTe/dec/Intrinsic_Defect/Te_i/random2/initial_
↪structure/q0
Failed job 245.host5: /home/test/CdTe/dec/Intrinsic_Defect/Cd_Te1/initial_structure/q0
...
```

It can be seen that there occur some mistakes in the structural relaxation for the neutral Cd_Te1 which leads to the calculation can not complete. But the program is not interrupted and will continue to finish the calculations except for Cd_Te. Therefore, users need to do nothing and wait for the program to complete. (the problems about defect Cd_Te will be solved after the program is completed.) If encountered the VASP error, please see **Common Problems and Solutions**.

Generate calculation directories for the charged defects:

After finishing the calculations of all the neutral defects (except for Cd_Te and interstitial with high energy), the program will judge the charge states of each defect and generate the corresponding directories and files for the charged defects based on the results of the neutral defects. A prompt will be given for those defects with calculation errors (undo, failed, and not converged) or without subsequent calculation (skip). The relevant information in `3dec.out` is as follows:

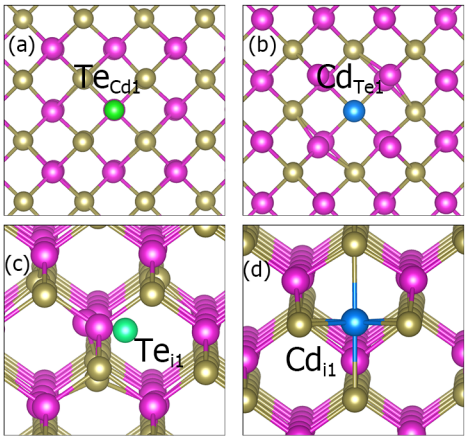
```
############ Ionized Defect module start ############
Start generating ionized defects
Warning: no EIGENVAL in /home/test/CdTe/dec/Intrinsic_Defect/Cd_Te1/initial_structure/q0/
↪static, skipped this directory!
Start generating ionized defects
Ionized defect path: /home/test/CdTe/dec/Intrinsic_Defect/V_Te1/initial_structure/q+1
Ionized defect path: /home/test/CdTe/dec/Intrinsic_Defect/V_Te1/initial_structure/q+2
Ionized defects generation completed
Start generating ionized defects
Ionized defect path: /home/test/CdTe/dec/Intrinsic_Defect/Te_i/random3/initial_structure/
↪q+1
Ionized defect path: /home/test/CdTe/dec/Intrinsic_Defect/Te_i/random3/initial_structure/
↪q+2
Ionized defect path: /home/test/CdTe/dec/Intrinsic_Defect/Te_i/random3/initial_structure/
↪q+3
Ionized defect path: /home/test/CdTe/dec/Intrinsic_Defect/Te_i/random3/initial_structure/
↪q+4
...
```

Take some defect structures to see in the visualization software, as shown in below:

Part of defect structures of CdTe.

Submit jobs for the defects with q 0:

After the structures and directories of the charged defects are generated, DEC module will call VASP to perform structural relaxation with PBE and total energy calculation with HSE (corresponds to the parameter *level = 2* in `dasp.in`). The waiting time needed in this step will be longer than that in 3.2.2. The messages in `3dec.out` are as follows:




```
############ AutoRun - Ionized Defect module start ############
Job 693.host5 submitted: /home/test/CdTe/dec/Intrinsic_Defect/V_Te1/initial_structure/q+2
Job 694.host5 submitted: /home/test/CdTe/dec/Intrinsic_Defect/V_Te1/initial_structure/q+1
Job 695.host5 submitted: /home/test/CdTe/dec/Intrinsic_Defect/Te_i/random3/initial_
↪structure/q+2
Job 696.host5 submitted: /home/test/CdTe/dec/Intrinsic_Defect/Te_i/random3/initial_
↪structure/q+1
Job 700.host5 submitted: /home/test/CdTe/dec/Intrinsic_Defect/Te_i/random3/initial_
↪structure/q+4
Succeed job 694.host5: /home/test/CdTe/dec/Intrinsic_Defect/V_Te1/initial_structure/q+1
Succeed job 693.host5: /home/test/CdTe/dec/Intrinsic_Defect/V_Te1/initial_structure/q+2
...
```

Calculate the correction for the charged defects:

After finishing the calculations of all the charged defects (except for Cd_Te), DEC module will calculate the FNV correction (according to the parameter *correction* = *FNV* in `dasp.in`), and then the formation energies and transition energy levels are also calculated. The specific data of the corrections and formation energies of different charge states of each defect are recorded in file `3dec.out` :

```
...
The formation energy (neutral) of V_Te1 at p1 is 1.684321
The formation energy (neutral) of V_Te1 at p2 is 2.869721
The FNV correction (q = 2) E_correct =  0.279795 eV
The transition level (0/2+) above VBM: 1.2429
The FNV correction (q = 1) E_correct =  0.087991 eV
The transition level (0/+) above VBM: 1.2833
...
```

All the data related to formation energies and transition energy levels are also recorded in the file `defect.log` under each defect's corresponding directories.

Output the image of formation energy:

Up to now, the program has already been completed, but we find the defect Cd_Te were not calculated by the output. The solution is as follows:

1. According to the error information, adjust the parameters in INCAR in the directory /home/test/CdTe/dec/Intrinsic_Defect/Cd_Te1/initial_structure/q0.

2. Return to the directory dec and create a new file named `redo.in` , and write /home/test/CdTe/dec/Intrinsic_Defect/Cd_Te1/initial_structure/q0.

3. Return to the directory CdTe and execute the DEC module again with the command `dasp 3` . The program will automatically judge the completed calculations, and recalculate the defect according to the `redo.in` .

4. The DEC module will carry out the calculations for the neutral and charged defect of Cd_Te, and calculate their formation energies.

Finally, DEC will automatically outputs the image of defect formation energy v.s. Fermi level by using all the corrected defect formation energies of CdTe at two chemical potentials. As shown in below:

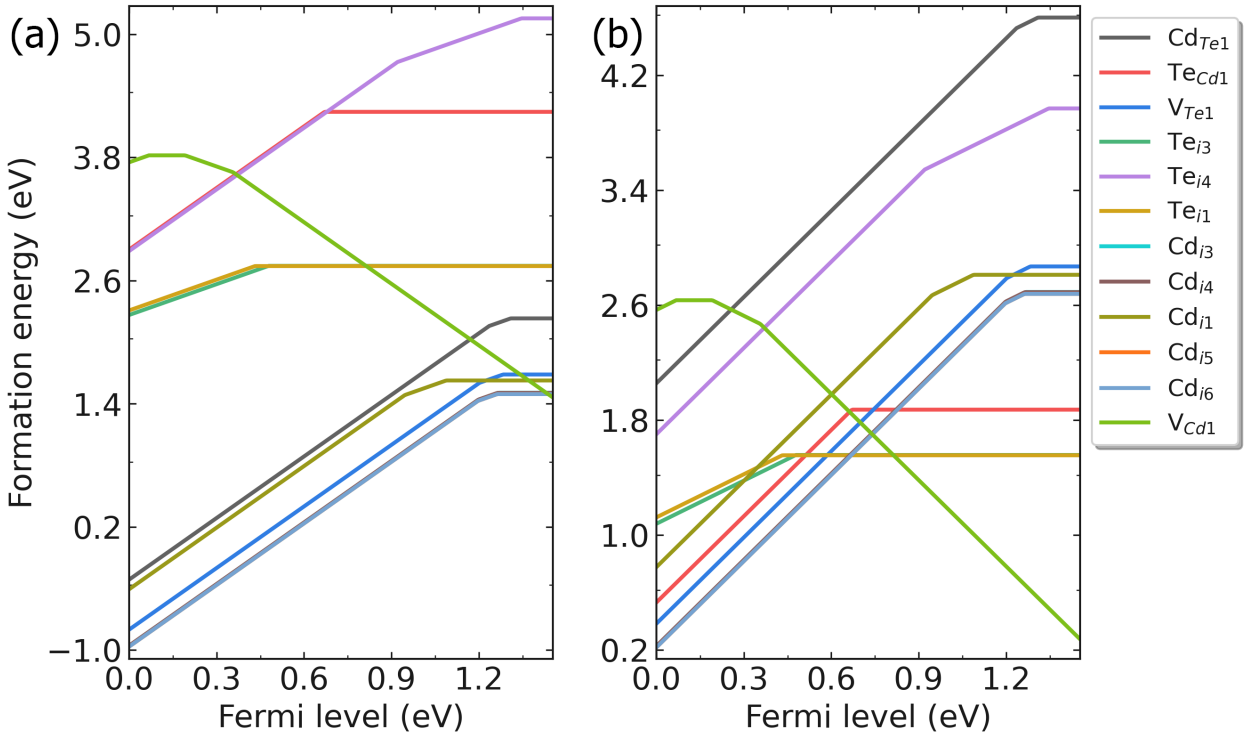
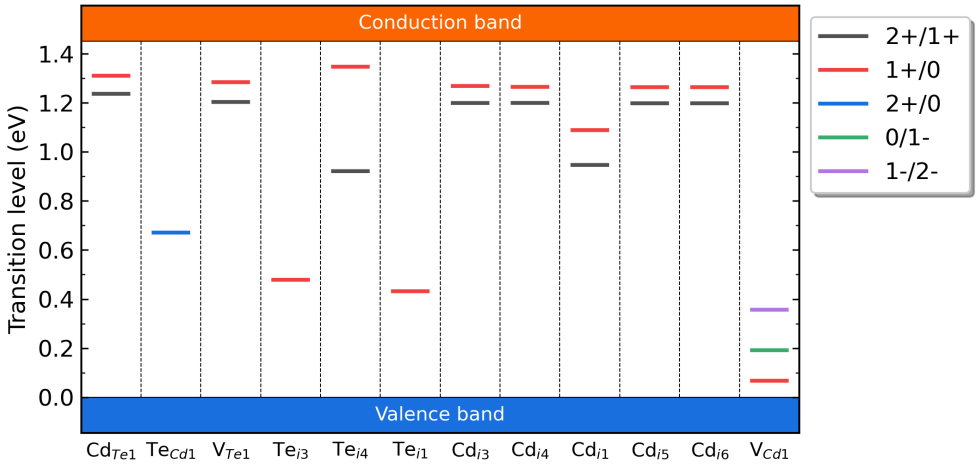


Fig: Formation energies of intrinsic defects in CdTe as functions of Fermi level under (a) Cd-rich and (b) Te-rich conditions.

Fig: The charge-state transition energy levels of intrinsic defects in CdTe.



5.1.4 5.1.4 DDC -- defect density and Fermi level calculations

5.1.4.1 Run DDC module

After finishing the DEC module, return to the directory CdTe and use the command `dasp 4` to execute the DDC module. No additional operation is required while waiting for the program to complete.

5.1.4.2 Workflow of DDC module

Firstly, DDC module will judge which defects have been finished calculating based on the output of DEC module, and take all these defects into consideration in the DDC calculation. Then, DDC will search for the output information about each defect as formation energy, transition energy levels, and degeneracy factor. Finally, summarize all the data and write in the file `DefectParams.txt` .

`4ddc.out` is the log file of DDC module:

```
############ Collecting information from DEC ############
Read defect types from DEC calculation successfully.
Defects considered in DDC calculation: ['Cd_Te1', 'Te_Cd1', 'V_Te1', 'Te_i-3', 'Te_i-4',
↪'Te_i-1', 'Cd_i-3', 'Cd_i-4', 'Cd_i-1', 'Cd_i-5', 'Cd_i-6', 'V_Cd1']
Chemical potentials change from p1 to p2.
Calculate gq for defect in each charge state.
Calculate Nsites for Cd_Te1: 1.373114e+22 cm^-3.
Calculate Nsites for Te_Cd1: 1.373114e+22 cm^-3.
Calculate Nsites for V_Te1: 1.373114e+22 cm^-3.
Calculate Nsites for Te_i-3: 9.154092e+21 cm^-3.
Calculate Nsites for Te_i-4: 9.154092e+21 cm^-3.
Calculate Nsites for Te_i-1: 9.154092e+21 cm^-3.
Calculate Nsites for Cd_i-3: 9.154092e+21 cm^-3.
Calculate Nsites for Cd_i-4: 9.154092e+21 cm^-3.
```

(continues on next page)

(continued from previous page)

```
Calculate Nsites for Cd_i-1: 9.154092e+21 cm^-3.
Calculate Nsites for Cd_i-5: 9.154092e+21 cm^-3.
Calculate Nsites for Cd_i-6: 9.154092e+21 cm^-3.
Calculate Nsites for V_Cd1: 1.373114e+22 cm^-3.
############ Collecting information from DEC ############
```

Below is the file `DefectParams.txt` :

```
1000 300
0.090000 0.840000
1.454001
Cd_Te1 1.373114e+22 1 1.3094 2 1.2726 1 x x x x x x x x x x x 2.230998 4.601798
Te_Cd1 1.373114e+22 1 0.4662 2 0.6708 1 0.3386 2 0.1484 1 x x x x x x x x 4.243870 1.
↪873070
V_Te1 1.373114e+22 1 1.2833 2 1.2429 1 x x x x x x x x x x x 1.684321 2.869721
Te_i-3 9.154092e+21 1 0.4786 2 0.153 1 0.1435 2 0.012 1 x x x x x x x x 2.743050 1.557650
Te_i-4 9.154092e+21 1 1.3461 2 1.1335 1 0.6062 2 0.4609 1 x x x x x x x x 5.154681 3.
↪969281
Te_i-1 9.154092e+21 1 0.432 2 0.1299 1 0.0022 2 -0.0831 1 x x x x x x x x 2.740914 1.
↪555514
Cd_i-3 9.154092e+21 1 1.2677 2 1.2331 1 0.7237 2 0.4498 1 x x x x x x x x 1.506642 2.
↪692042
Cd_i-4 9.154092e+21 1 1.2641 2 1.2314 1 0.7192 2 0.4466 1 x x x x x x x x 1.505396 2.
↪690796
Cd_i-1 9.154092e+21 1 1.0881 2 1.0171 1 0.5865 2 0.3569 1 x x x x x x x x 1.626088 2.
↪811488
Cd_i-5 9.154092e+21 1 1.2629 2 1.2301 1 0.7187 2 0.4461 1 x x x x x x x x 1.493426 2.
↪678826
Cd_i-6 9.154092e+21 1 1.2629 2 1.2301 1 0.7187 2 0.4461 1 x x x x x x x x 1.493426 2.
↪678826
V_Cd1 1.373114e+22 1 0.0676 2 0.0115 1 -0.0427 2 -0.0964 1 0.1918 2 0.2742 1 0.7753 2 1.
↪0413 1 3.819933 2.634533
```

Self-consistent calculation under growth temperature:

The DDC module will calculate the defect/dopant and carrier densities at the temperature T=1000 K (*ddc_temperature = 1000 300*), and obtain the Fermi level by self-consistently under the charge neutralization condition.

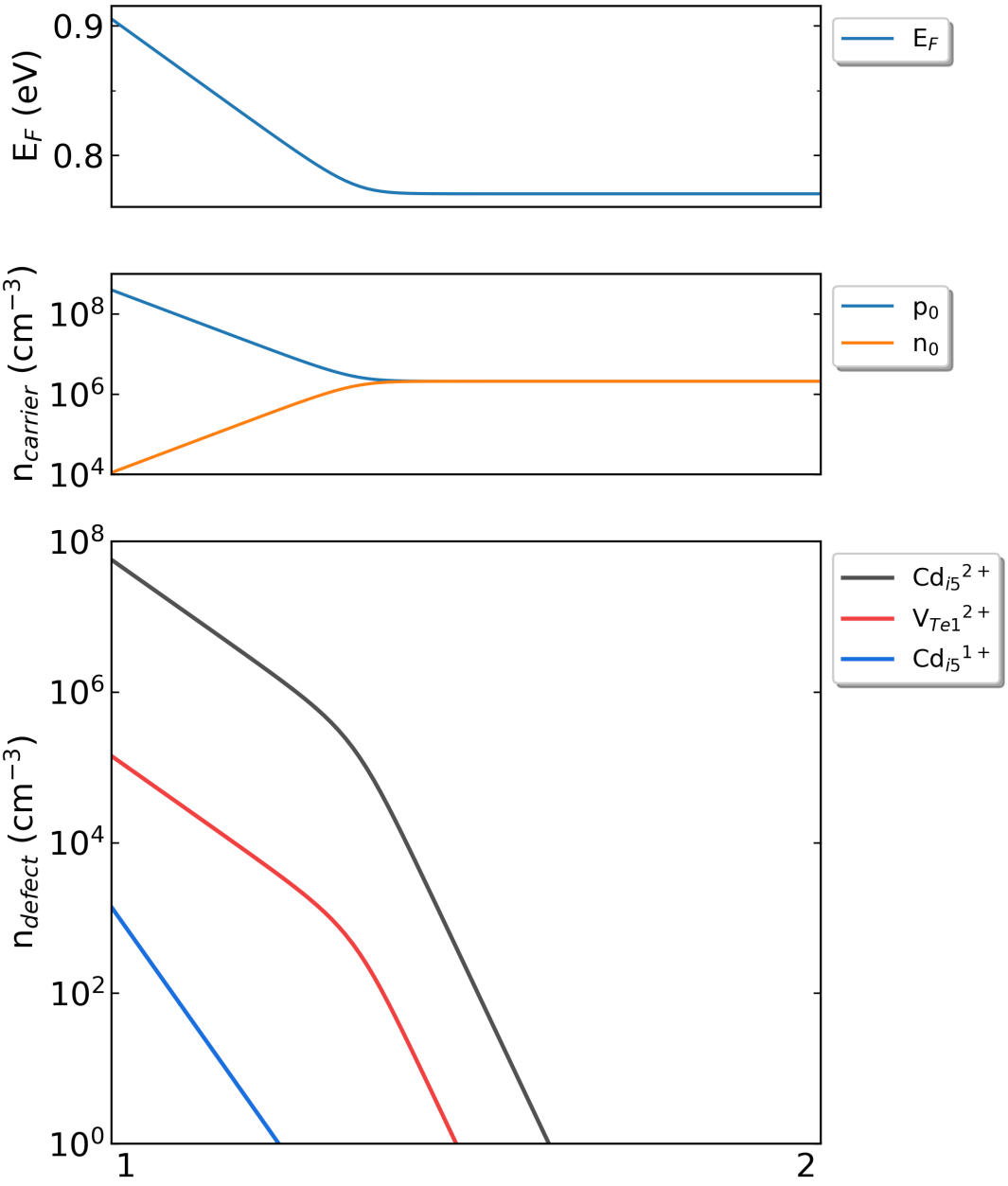
Self-consistent calculation under working (measuring) temperature:

The DDC module will recalculate the defect/dopant and carrier densities at the temperature T=300 K (*ddc_temperature = 1000 300*), and re-obtain the Fermi level by self-consistently under the charge neutralization condition.

Output defect density:

DDC module will output four files in the directory CdTe/ddc: `Fermi.dat` , `Carrier.dat` , `Defect_charge.dat` , and `density.png` . Using different growth temperatures, the images `density.png` are as follows:

Fig: The Fermi level, electron and hole carrier densities, and defect densities in CdTe as functions of the chemical potentials with the growth temperature is 300 K and the working temperature is 300 K.


0.9
0.8
E_F (eV)
E_F
10^8
10^6
10^4
$n_{carrier}$ (cm^{-3})
p_0
n_0
10^8
10^6
10^4
10^2
10^0
n_{defect} (cm^{-3})
Cd_{i5}^{2+}
V_{Te1}^{2+}
Cd_{i5}^{1+}
1
2

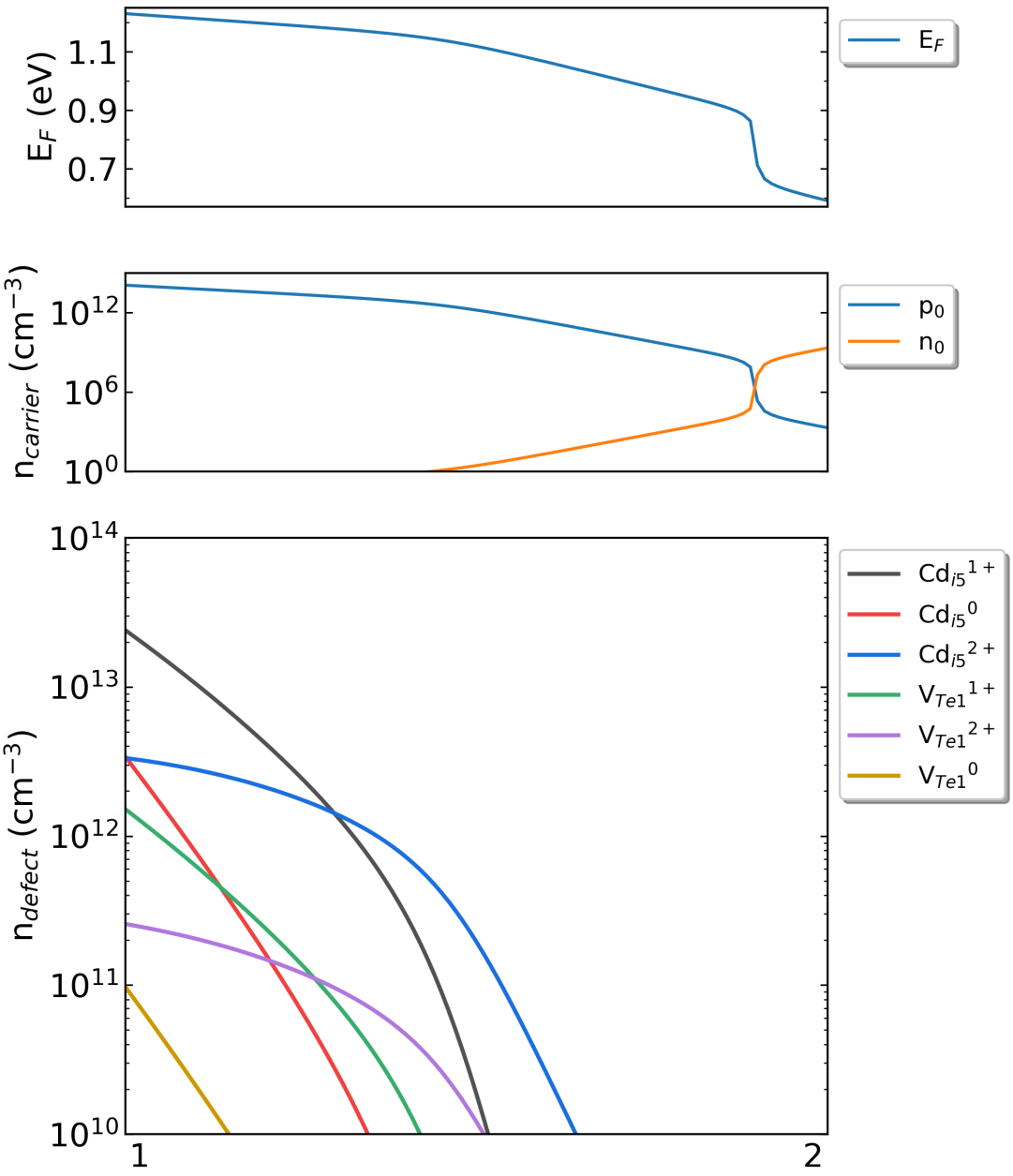

E_F (eV)
1.1
0.9
0.7
E_F
n_carrier (cm^-3)
10^12
10^6
10^0
p_0
n_0
n_defect (cm^-3)
10^14
10^13
10^12
10^11
10^10
1
2
Cd_i5^1+
Cd_i5^0
Cd_i5^2+
V_Te1^1+
V_Te1^2+
V_Te1^0

Fig: The Fermi level, electron and hole carrier densities, and defect densities in CdTe as functions of the chemical potentials with the growth temperature is 600 K and the working temperature is 300 K.

Fig: The Fermi level, electron and hole carrier densities, and defect densities in CdTe as functions of the chemical potentials with the growth temperature is 1000 K and the working temperature is 300 K.

5.2 5.2 The calculations of intrinsic defects in HfO2

HfO2 is an important material widely used in electronic devices with a high dielectric constant. Recently, it has attracted a lot of attention as its metastable phase possesses ferroelectric. The defects in HfO2 may affect its phase stability, and dielectric and ferroelectric properties, so it is necessary to systematically calculate the defect properties of HfO2.

With the help of the efficient software DASP, the properties of the point defects and the relationship between the defects formation and its phase stability in HfO2 can be systematically studied.

The following is an example of using DASP to calculate intrinsic defects in HfO2:

5.2.1 5.2.1 PREPARE --- Prepares for calculation

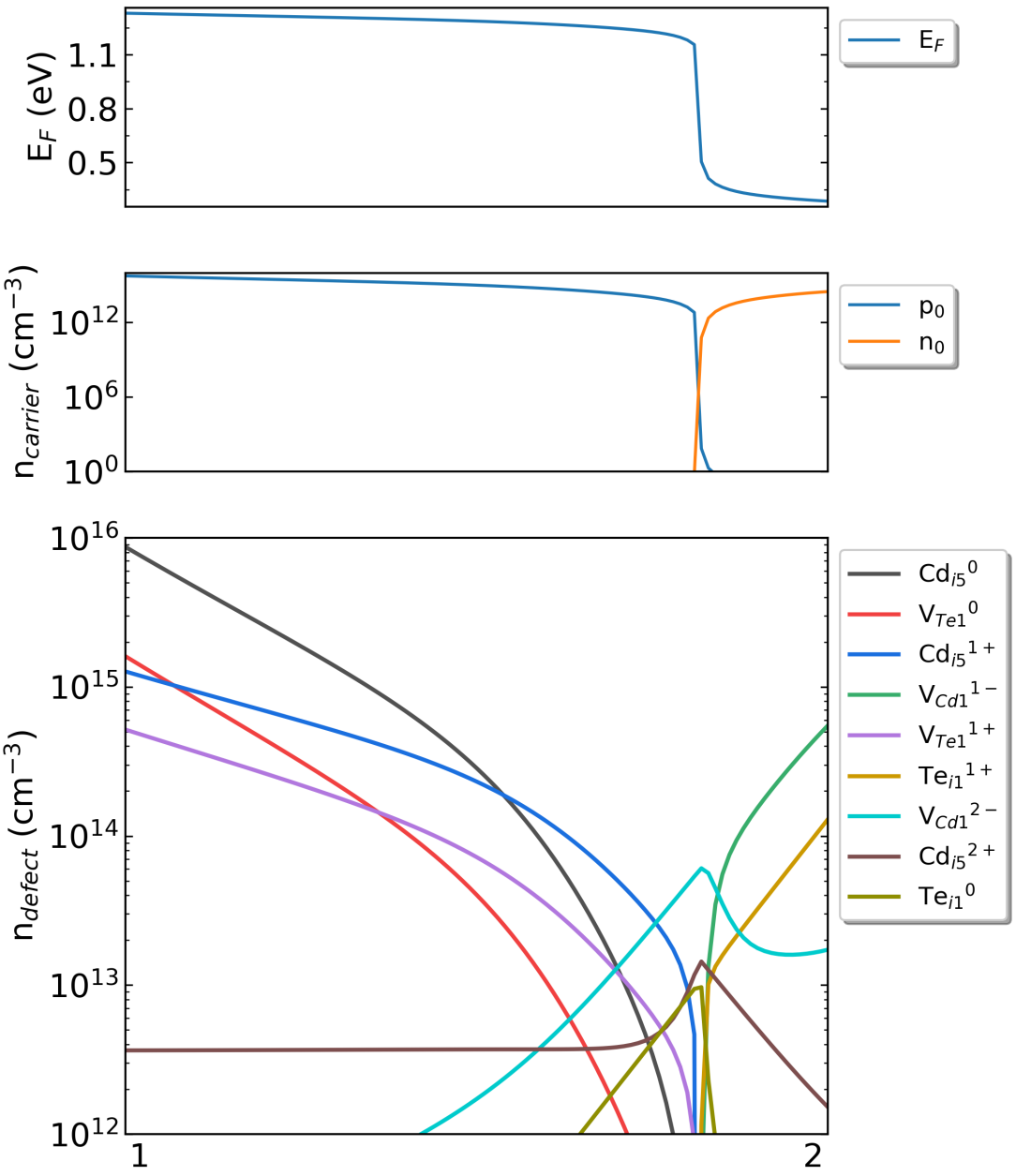
5.2.1.1 POSCAR and dasp.in

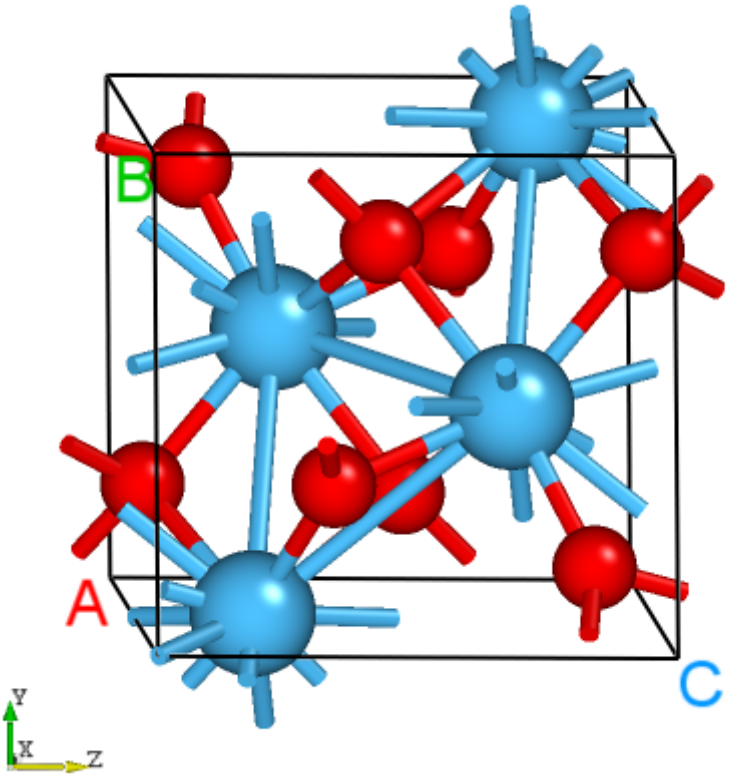
Find the `POSCAR` for HfO2 from the `Materials Project` database, as follows:

```
 Hf4 O8
    1.00000000000000
         5.0652456351012756    0.0000000000000000   -0.8648023964655065
         0.0000000000000000    5.1942160710694010    0.0000000000000000
        -0.0006123763055649    0.0000000000000000    5.3264852554835196
Hf   O
4    8
Direct
   0.7239198560286844  0.5430528338948646  0.2919471319794364
   0.2760801439713155  0.0430528338948647  0.2080528680205637
   0.2760801439713155  0.4569471661051352  0.7080528680205705
   0.7239198560286844  0.9569471661051354  0.7919471319794295
   0.5514108623083260  0.2575137054162841  0.0226462276199697
   0.4485891376916739  0.7575137054162840  0.4773537723800304
   0.4485891376916739  0.7424862945837160  0.9773537723800304
   0.5514108623083260  0.2424862945837159  0.5226462276199696
   0.9317881306845183  0.6693565882783469  0.6530354073676818
   0.0682118693154819  0.1693565882783468  0.8469645926323182
   0.0682118693154819  0.3306434117216532  0.3469645926323183
   0.9317881306845183  0.8306434117216531  0.1530354073676817
```

Taking it into the crystal visualization software, the structure can be seen as below:

The structure of HfO2.


E_F (eV)
1.1
0.8
0.5
E_F
n_carrier (cm^-3)
10^12
10^6
10^0
p0
n0
n_defect (cm^-3)
10^16
10^15
10^14
10^13
10^12
1
2
Cd_i5^0
V_Te1^0
Cd_i5^1+
V_Cd1^1-
V_Te1^1+
Te_i1^1+
V_Cd1^2-
Cd_i5^2+
Te_i1^0



Next, use VASP to optimize its lattice parameters or modify the lattice to match the experimental measurements. Users need to do it manually.

Write the required parameters in `dasp.in` :

```
############### Job Scheduling ###############
cluster = PBS      # (job scheduling system)
node_number = 1          # (number of node)
core_per_node = 96       # (core per node)
queue = batch            # (name of queue/partition)
max_time = 24:00:00         # (maximum time for a single DFT calculation)
vasp_path_dec = /opt/vasp.5.4.4/bin/vasp_gam    #  (path of VASP)
vasp_path_tsc = /opt/vasp.5.4.4/bin/vasp_std
job_name = submit_job     # (name of script)
potcar_path = /opt/POT/potpaw_PBE     # (path of pseudopotentials)
max_job = 5

############### TSC Module ###############
database_api = ******************** # (str-list type)

############### DEC Module ###############
level = 2    # (level=1: PBE+PBE; level=2: PBE+HSE; level=3: HSE+HSE)
```

(continues on next page)

(continued from previous page)

```
min_atom = 96
max_atom = 96
intrinsic = T   # (default: T)
correction = FNV   # (default: none)
epsilon = 10.3
Eg_real = 1.45   # (experimental band gap)

############## DDC Module ##############
ddc_temperature = 800 300
ddc_mass = 2.95 2.99
```

Next, all the parameters listed in `dasp.in` will be described:

```
cluster = PBS
# The system of the used cluster is PBS.
```

```
node_number = 1
# One node is used for each calculation.
```

```
core_per_node = 96
# 96 cores are used for each node, so 1*96=96 cores are used in total for each␣
↪calculation.
```

```
queue = batch
# The queue named "batch" is used to carry out calculations. Therefore, users need to␣
↪make sure the queue name, nodes, and cores of clusters before configuring dasp.in.
```

```
max_time = 24:00:00        # (maximum time for a single DFT calculation)
# The maximum time allowed is 24 hours for a single DFT calculation and can be set␣
↪arbitrarily.
```

```
vasp_path_dec = /opt/vasp.5.4.4/bin/vasp_gam   #  (path of VASP)
vasp_path_tsc = /opt/vasp.5.4.4/bin/vasp_std
# The VASP_std version is used for TSC calculations, and the VASP_gam version is used␣
↪for DEC calculations.
```

```
job_name = submit_job    # (name of script)
# The submission script, named " submit_job" and can be set arbitrarily.
```

```
potcar_path = /opt/POT/potpaw_PBE    # (path of pseudopotentials)
# path of pseudopotentials
```

```
max_job = 5
# the allowed maximum number of jobs at the same time
```

```
database_api = ******************* # (str-list type)
# using to visit the Materials Project database
```

```
level = 2   # (level=1: PBE+PBE; level=2: PBE+HSE; level=3: HSE+HSE)
# using GGA-PBE for structural relaxation and HSE to calculate the total energy
```

```
min_atom = 96
max_atom = 96
# The number of atoms within the generated supercell that we want is 96, and as far as␣
↪possible to make a=b=c and a b c.
```

```
intrinsic = T   # (default: T)
# Generate intrinsic defects, V_Hf, V_O, Hf_O, O_Hf, Hf_i, and O_i.
```

```
correction = FNV   # (default: None)
# Generate intrinsic defects, V_Hf, V_O, Hf_O, O_Hf, Hf_i, and O_i.
```

```
epsilon = 21.6
# The dielectric constant of HfO2 is 21.6.
```

```
Eg_real = 5.68   # (experimental band gap)
# The experimental band gap of HfO2 is about 5.68 eV, DASP will adjust AEXX in INCAR to␣
↪make the band gap of the supercell without defect equal to 5.68 eV.
```

```
ddc_temperature = 1000 300
# the growth temperature set to 1000 K and the working temperature set to 300 K.
```

```
ddc_mass = 2.95 2.99
# electron effective mass set to 2.95 and hole effective mass set to 2.99.
```

5.2.1.2 Use DASP to generate the required input files

Create a new directory HfO2, then prepare the files, `POSCAR` and `dasp.in` , mentioned above in the directory ./HfO2/. Next, execute `dasp 1` to start PREPARE module and no additional operation is needed thereafter. DASP will output file `1prepare.out` to record the running log of the module.

5.2.1.3 Workflow of PREPARE module

Generate supercell:

Firstly, the program will automatically find the optimal supercell (as far as possible to make a=b=c and a b c) based on the parameters min_atom=96 and max_atom=96 and output POSCAR for the supercell. The following are the structure messages of HfO2 supercell `POSCAR_nearlycube` .

```
Cubic_cell
1.0
10.2770806222 0.0000000000 0.0000000000
0.0000000000 10.3884321420 0.0000000000
-1.7940733247 0.0000000000 10.5008134501
```

(continues on next page)

(continued from previous page)

```
Hf O
32 64
Direct
0.3619599280 0.2715264169 0.1459735659
0.1380400719 0.0215264169 0.1040264340
0.1380400719 0.2284735830 0.3540264340
0.3619599280 0.4784735830 0.3959735659

...
```

Taking it into the crystal visualization software, the supercell can be seen as below:

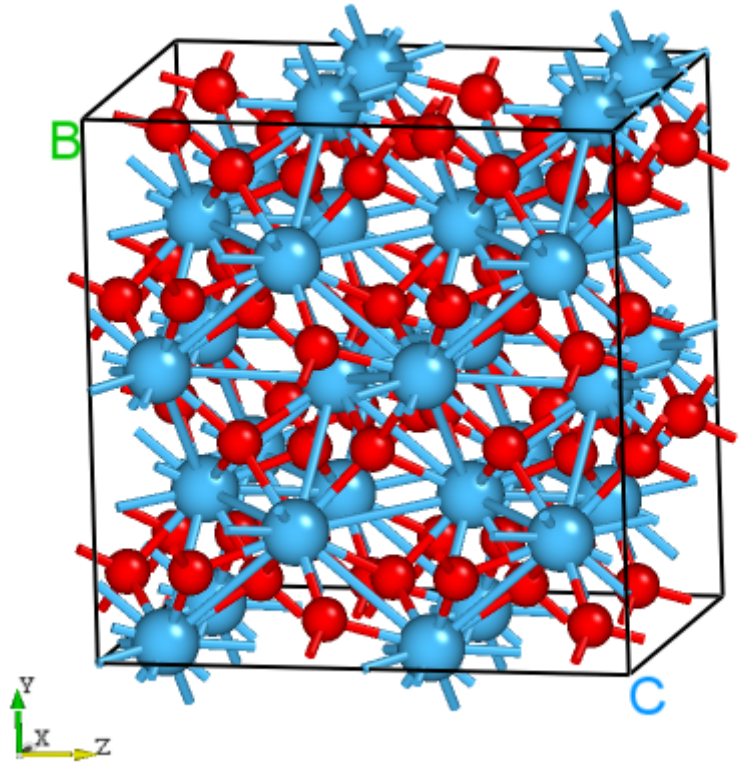


Fig: The structure of HfO2 supercell generated by DASP.

Madelung constant calculation:

Secondly, according to the generated supercell file, the program will execute Madelung constant calculation which describes the Coulomb interaction between charged defect and periodic image charge.

After finishing the above two steps calculation, the output of `1prepare.out` is as follows:

```
############ Prepare Files module start ############

Read the structure file POSCAR you provided
Get the refined cell POSCAR_refined from POSCAR
Generate the nearlycube cell POSCAR_nearlycube from POSCAR
Generate job script through dasp.in parameters
Generate single-point KPOINTS
Generate pseudopotential file POTCAR through potcar_dir you set
Generate commonly used vasp input file INCAR
Start the madelung constant calculation
Generate the madelung calculation directory
Generate madelung calculation POSCAR
Generate madelung calculation POTCAR
Generate madelung calculation INCAR
Generate madelung calculation KPOINTS
Generate madelung calculation job script
Job 103.host5 submitted: /home/fudan/HfO2/dec/madelung/static
Succeed job 103.host5: /home/fudan/HfO2/dec/madelung/static
The madelung constant calculation completed
The madelung constant = 2.841
```

HSE exchange proportion calculation:

According to the generated supercell file, the program will perform HSE static calculations with AEXX=0.25 and AEXX=0.3 respectively to determine the value of AEXX which can make the obtained band gap match *Eg_real = 5.68* based on the slope. If the calculated band gap with AEXX=0.25 or AEXX=0.3 is consistent with the set, the subsequent calculation for AEXX will not be performed. This is the example where the bandgap is satisfied when AEXX=0.25. Therefore, after the calculation is completed, the contents in the directory HfO2/dec/AEXX/ are as follows:

```
cd ./dec/AEXX
ls
0.25  AEXX.list
```

It indicates that the band gap of the HfO2 supercell is 5.68 eV when AEXX=0.25 (two decimal places), and write this parameter into INCAR. Meanwhile, the log can be seen from `1prepare.out` as follows:

```
Start the HSE parameter AEXX calculation
Job 107.host5 submitted: /home/fudan/HfO2/dec/AEXX/0.25/static
Succeed job 107.host5: /home/fudan/HfO2/dec/AEXX/0.25/static
The HSE parameter AEXX calculation completed
The HSE parameter AEXX = 0.25
level = 2: Generate PBE relax vasp input file INCAR-relax
level = 2: Generate HSE static vasp input file INCAR-static
```

Optimize the ionic position of the host supercell:

The last step in PREPARE module is to optimize the ionic position of the host supercell according to level=2 (i.e. PBE relax). The optimized file is `POSCAR_final` in the directory HfO2/dec/relax. At the same time, the sign of the end of DASP operation can be seen in `1prepare.out` , and it also tells us that we need to do the TSC module calculation in the next step.

```
Start the POSCAR_nearlycube relax calculation
Generate the POSCAR_nearlycube relax directory
Job 110.host5 submitted: /home/fudan/HfO2/dec/relax
Succeed job 110.host5: /home/fudan/HfO2/dec/relax
The POSCAR_nearlycube relax calculation completed
Get the final structure POSCAR_final

############ Prepare Files module end ############

DASP-PREPARE finished, please run DASP-TSC next
```

5.2.2 5.2.2 TSC -- thermodynamic stability and chemical potential calculations

5.2.2.1 Run TSC module

The directory HfO2/dec will be created when using the command *dasp 1* to execute PREPARE module, and generate file `1prepare.out` in this directory. After finishing the program, there has the corresponding completion flag in `1prepare.out` . Then, enter the directory HfO2/dec and confirm that the parameters in `INCAR-relax` and `INCAR-static` are feasible. (Users can modify INCAR, and DASP will make subsequent calculations based on the INCAR in this directory.)

Once confirm that the PREPARE module is finished, return to the directory HfO2 and use the command *dasp 2* to execute the TSC module. Similarly, the TSC module will create a directory named tsc under the directory HfO2, which contains the output of the TSC program, including every calculation directory and the running log file `2tsc.out` . No additional operation is required while waiting for the program to complete.

5.2.2.2 Workflow of TSC module

The total energy calculation of the host structure (the parameters are consistent with MP database):

TSC module will use the same input parameters (INCAR, KPOINTS, POTCAR) with the `Materials Project` database to perform structural relaxation and static calculation on the primitive cells given by the user. Therefore, the calculated total energy is comparable to that of the MP database. This step is to obtain the **key hetero-phases** that limit the stability of HfO2. In the directory, we can see:

```
cd tsc
cd HfO2/
ls
relaxation1  relaxation2  static
```

The running log also can be seen from the HfO2/tsc/2tsc.out, that is, the steps such as generating input files, relaxation1, relaxation2, static and data extraction.

The judgement of key hetero-phases compounds:

The TSC module will search for all the secondary compounds that compete with HfO2 in the MP database. And compare the total energy of HfO2 calculated in the previous step with that of the hetero-phases extracted from the database to confirm HfO2 is **thermodynamically stable**.
Subsequently, the program will automatically download the key hetero-phase compounds that can limit the thermodynamic stability of HfO2. Only Hf and O2 are considered in this case. The relevant information can be seen in `2tsc.out` :

```
...
analysing the thermodynamic stability of HfO2.
key phases of HfO2 are: Hf O2 .
file key_phases_info_recalc.yaml generated.
analysing of HfO2 is done.
...
```

The total energy calculation of the host and hetero-phase compounds:

After the key hetero-phase compounds are confirmed, TSC will calculate the total energy of HfO2, Hf, and O2 by using the parameter (AEXX) obtained from PREPARE module. `2tsc.out` is as follows:

```
...
Job 182.host5 submitted: /home/test/HfO2/tsc/HfO2/static_recalc
Job 183.host5 submitted: /home/test/HfO2/tsc/Hf/static_recalc
Job 184.host5 submitted: /home/test/HfO2/tsc/O2/static_recalc
Succeed job  182.host5: /home/test/HfO2/tsc/HfO2/static_recalc
Succeed job  183.host5: /home/test/HfO2/tsc/Hf/static_recalc
Succeed job  184.host5: /home/test/HfO2/tsc/O2/static_recalc
...
```

The chemical potential calculation:

Calculating the formation energy and stable chemical potential region of HfO2 based on the calculated total energy. As HfO2 is a binary compound, TSC module will give the endpoint of two chemical potentials, i.e. Hf-rich and O-rich, and write them into `dasp.in` :

```
# The orders are consistent with the order of elements in POSCAR, i.e. the first column␣
↪is Hf and the second column is O.
E_pure = -11.1092 -8.2689
p1 = 0.0 -5.8748
p2 = -11.7496 0.0
```

The output after the program is completed can be seen in `2tsc.out` :

```
dir '2d-figures','3d-figures','ori_data_MP' ready.  try to read file: 'calc_list.yaml'.
analysing the thermodynamic stability of HfO2.
key phases of HfO2 are: Hf O2 .
analysing of HfO2 is done.
---------------------------
DASP-TSC finished
```

For the ternary and multinary compounds, TSC module will output the image of the stable region and the chemical potential at the endpoint of the stable region.

5.2.3 5.2.3 DEC -- the calculations of defect formation energy and transition energy level

5.2.3.1 Run DEC module

The directory HfO2/tsc will be created when using the command *dasp 2* to execute TSC module, and generate file `2tsc.out` in this directory. After finishing the TSC module, there has the corresponding completion flag in `2tsc.out` . Then, open the file `dasp.in` under the directory HfO2/dasp.in to confirm the chemical potential already has been written.

Once confirm that the TSC module is finished, return to the directory HfO2 and use the command *dasp 3* to execute the DEC module. DEC will output relevant files in the generated directory dec in the first step, including the defect structures, directories, and the log file `3dec.out` . No additional operation is required while waiting for the program to complete.

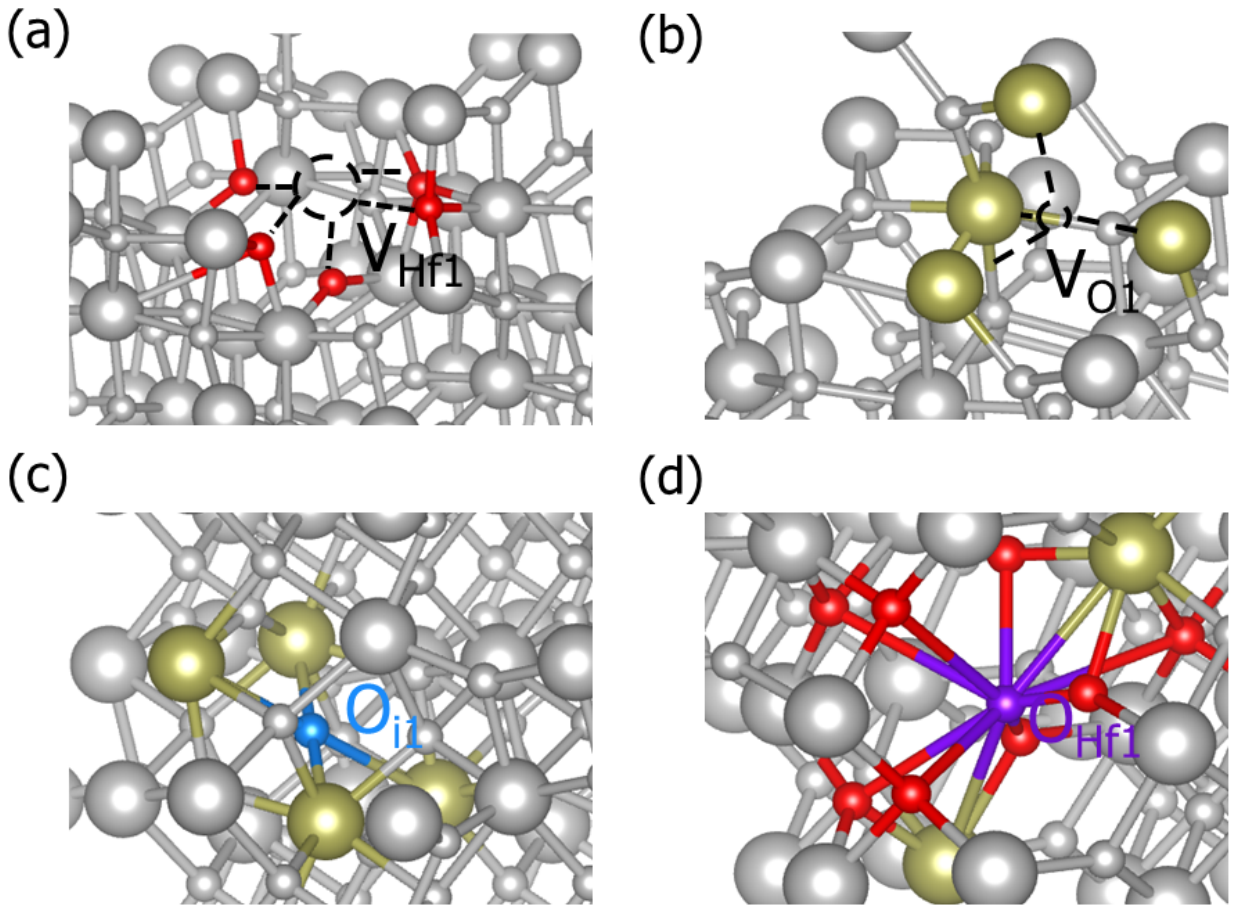
5.2.3.2 Workflow of DEC module

Generate defect structure:

Based on the parameter intrinsic = T in `dasp.in` , DEC will generate the intrinsic defects for HfO2, i.e. create the calculation directory HfO2e/dec/Intrinsic_Defect, in which the structures and directories of vacancies V_Hf and V_O, antisite defects Hf_O and O_Hf, as well as interstitial defects Hf_i and O_i are included. According to the crystal symmetry analysis, there has no inequivalent site for Hf atom, but two inequivalent sites for O atom. Therefore, there will generate two different configurations for V_O and Hf_O and one configuration for V_Hf and O_Hf, while the number of configurations for Hf_i and O_i is depended on the input parameter set by users.

```
cd dec/Intrinsic_Defect/
ls
Hf_i  Hf_O1  Hf_O2  host  Intrinsic_Defect.list  O_Hf1  O_i  V_Hf1  V_O1  V_O2
```

Take some defect structures to see in the visualization software, as shown in below:


(a)
V_{Hf1}
(b)
V_{O1}
(c)
O_{i1}
(d)
O_{Hf1}

Part of defect structures of HfO2.

At the same time, the output of DEC module can be seen in `3dec.out` as follows:

```
############ Neutral Defect module start ############

Make intrinsic defect directory Intrinsic_Defect
Generate host directory in Intrinsic_Defect
Start generating neutral vacancy defect
Generate neutral defect at: V_Hf1/initial_structure/q0
Generate neutral defect at: V_O1/initial_structure/q0
Generate neutral defect at: V_O2/initial_structure/q0
Neutral vacancy defect generation completed
Start generating neutral intrinsic antisite defect
Generate neutral defect at: O_Hf1/initial_structure/q0
Generate neutral defect at: Hf_O1/initial_structure/q0
Generate neutral defect at: Hf_O2/initial_structure/q0
Neutral intrinsic antisite defect generation completed
Start generating neutral intrinsic interstitial defect
Generate neutral defect at: Hf_i/random1/initial_structure/q0
Generate neutral defect at: Hf_i/random2/initial_structure/q0
Generate neutral defect at: Hf_i/random3/initial_structure/q0
Generate neutral defect at: Hf_i/random4/initial_structure/q0
Generate neutral defect at: Hf_i/random5/initial_structure/q0
Generate neutral defect at: Hf_i/random6/initial_structure/q0
Generate neutral defect at: O_i/random1/initial_structure/q0
Generate neutral defect at: O_i/random2/initial_structure/q0
Generate neutral defect at: O_i/random3/initial_structure/q0
Generate neutral defect at: O_i/random4/initial_structure/q0
Generate neutral defect at: O_i/random5/initial_structure/q0
Generate neutral defect at: O_i/random6/initial_structure/q0
Neutral intrinsic interstitial defect generation completed

############ Neutral Defect module end ############
```

It can be seen that the DEC module only creates the directories for all the neutral defects at this time.

Submit jobs for all defects with q=0:

After the structures and directories of neutral defects are generated, DEC module will call VASP to perform structural relaxation with PBE and total energy calculation with HSE (corresponds to the parameter level=2 in `dasp.in`), this step may need a long time. Users can check the file `3dec.out` at any time. The messages in `3dec.out` are as follows:

```
Job 198.host5 submitted: /data/HfO2/dec/Intrinsic_Defect/V_O2/initial_structure/q0
Job 200.host5 submitted: /data/HfO2/dec/Intrinsic_Defect/V_O1/initial_structure/q0
```

(continues on next page)

(continued from previous page)

```
Job 202.host5 submitted: /data/HfO2/dec/Intrinsic_Defect/O_Hf1/initial_structure/q0
Job 204.host5 submitted: /data/HfO2/dec/Intrinsic_Defect/Hf_O1/initial_structure/q0
Job 206.host5 submitted: /data/HfO2/dec/Intrinsic_Defect/Hf_i/random5/initial_structure/
↪q0

...

Succeed job 202.host5: /data/HfO2/dec/Intrinsic_Defect/O_Hf1/initial_structure/q0
Succeed job 198.host5: /data/HfO2/dec/Intrinsic_Defect/V_O2/initial_structure/q0
Failed job 204.host5: /data/HfO2/dec/Intrinsic_Defect/Hf_O1/initial_structure/q0
Succeed job 206.host5: /data/HfO2/dec/Intrinsic_Defect/Hf_i/random5/initial_structure/q0

...
```

It can be seen that there occur some mistakes in the structural relaxation for the neutral Hf_O1 which leads to the calculation can not complete. But the program is not interrupted and will continue to finish the calculations except for Hf_O1. Therefore, users need to do nothing and wait for the program to complete. (the problems about defect Hf_O will be solved after the program is completed.) If encountered the VASP error, please see **Common Problems and Solutions**.

Generate calculation directories for the charged defects:

After finishing the calculations of all the neutral defects (except for Hf_O and interstitial with high energy), the program will judge the charge states of each defect and generate the corresponding directories and files for the charged defects based on the results of the neutral defects. A prompt will be given for those defects with calculation errors (undo, failed, and not converged) or without subsequent calculation (skip). The relevant information in `3dec.out` is as follows:

```
############ Ionized Defect module start ############
Start generating ionized defects
Ionized defect path: /data/HfO2/dec/Intrinsic_Defect/Hf_O2/initial_structure/q+1
Ionized defect path: /data/HfO2/dec/Intrinsic_Defect/Hf_O2/initial_structure/q+2
Ionized defect path: /data/HfO2/dec/Intrinsic_Defect/Hf_O2/initial_structure/q+3
Ionized defect path: /data/HfO2/dec/Intrinsic_Defect/Hf_O2/initial_structure/q+4
Ionized defects generation completed
Start generating ionized defects
Ionized defect path: /data/HfO2/dec/Intrinsic_Defect/V_O2/initial_structure/q-2
Ionized defect path: /data/HfO2/dec/Intrinsic_Defect/V_O2/initial_structure/q-1
Ionized defect path: /data/HfO2/dec/Intrinsic_Defect/V_O2/initial_structure/q+1
Ionized defect path: /data/HfO2/dec/Intrinsic_Defect/V_O2/initial_structure/q+2
Ionized defects generation completed
Warning: static calculation undo in /data/HfO2/dec/Intrinsic_Defect/Hf_O1/initial_
↪structure/q0/static, skipped generate ionized defect
The static calculation of /data/HfO2/dec/Intrinsic_Defect/Hf_i/random5/initial_structure/
↪q0/static is skipped, skip ionized defect generation

...
```

Submit jobs for the defects with q 0:

After the structures and directories of the charged defects are generated, DEC module will call VASP to perform structural relaxation with PBE and total energy calculation with HSE (corresponds to the parameter level=2 in `dasp.in`). The waiting time needed in this step will be longer than that in 3.2.2. The messages in `3dec.out` are as follows:

```
############ AutoRun - Ionized Defect module start ############
Job 259.host5 submitted: /data/HfO2/dec/Intrinsic_Defect/Hf_O2/initial_structure/q+4
Job 261.host5 submitted: /data/HfO2/dec/Intrinsic_Defect/Hf_O2/initial_structure/q+1
Job 263.host5 submitted: /data/HfO2/dec/Intrinsic_Defect/Hf_O2/initial_structure/q+3
Job 265.host5 submitted: /data/HfO2/dec/Intrinsic_Defect/Hf_O2/initial_structure/q+2
Job 267.host5 submitted: /data/HfO2/dec/Intrinsic_Defect/V_O2/initial_structure/q+1

...

Succeed job 259.host5: /data/HfO2/dec/Intrinsic_Defect/Hf_O2/initial_structure/q+4
Succeed job 261.host5: /data/HfO2/dec/Intrinsic_Defect/Hf_O2/initial_structure/q+1
Succeed job 263.host5: /data/HfO2/dec/Intrinsic_Defect/Hf_O2/initial_structure/q+3
Succeed job 267.host5: /data/HfO2/dec/Intrinsic_Defect/V_O2/initial_structure/q+1
Succeed job 265.host5: /data/HfO2/dec/Intrinsic_Defect/Hf_O2/initial_structure/q+2

...
```

Calculate the correction for the charged defects:

After finishing the calculations of all the charged defects (except for Hf_O1), DEC module will calculate the FNV correction (according to the parameter *correction = FNV* in `dasp.in`), and then the formation energies and transition energy levels are also calculated. The specific data of the corrections and formation energies of different charge states of each defect are recorded in file `3dec.out` :

```
...
The formation energy (neutral) of Hf_O2 at p1 is 5.339603
The formation energy (neutral) of Hf_O2 at p2 is 22.964003
The FNV correction (q = 4) E_correct =  1.80575 eV
The transition level (0/4+) above VBM: 3.9438
The FNV correction (q = 1) E_correct =  0.159401 eV
The transition level (0/+) above VBM: 4.7144
The FNV correction (q = 3) E_correct =  1.02256 eV
The transition level (0/3+) above VBM: 4.2496
The FNV correction (q = 2) E_correct =  0.502853 eV
The transition level (0/2+) above VBM: 4.4441
...

The static calculation of /data/HfO2/dec/Intrinsic_Defect/O_i/random4/initial_structure/
↪q0/static is skipped, skip formation energy calculation
Warning: calculation undo in /data/HfO2/dec/Intrinsic_Defect/Hf_O1/initial_structure/q0/
↪static, skipped calculate formation energy
```

(continues on next page)

(continued from previous page)

```
...
```

All the data related to formation energies and transition energy levels are also recorded in the file `defect.log` under each defect's corresponding directories.

Output the image of formation energy:

Up to now, the program has already been completed, but we find the defect Hf_O1 were not calculated by the output. The solution is as follows:

1. According to the error information, adjust the parameters in INCAR in the directory /home/test/HfO2/dec/Intrinsic_Defect/Hf_O1/initial_structure/q0.
2. Return to the directory dec and create a new file named `redo.in` , and write /home/test/HfO2/dec/Intrinsic_Defect/Hf_O1/initial_structure/q0.
3. Return to the directory HfO2 and execute the DEC module again with the command *dasp 3* . The program will automatically judge the completed calculations, and recalculate the defect according to the `redo.in` .
4. The DEC module will carry out the calculations for the neutral and charged defect of Hf_O1, and calculate their formation energies.

Finally, DEC will automatically outputs the image of defect formation energy v.s. Fermi level by using all the corrected defect formation energies of HfO2 at two chemical potentials. As shown in below:

Fig: Formation energies of intrinsic defects in HfO2 as functions of Fermi level at p1 point (Hf-rich condition).

Fig: Formation energies of intrinsic defects in HfO2 as functions of Fermi level at p2 point (O-rich condition).

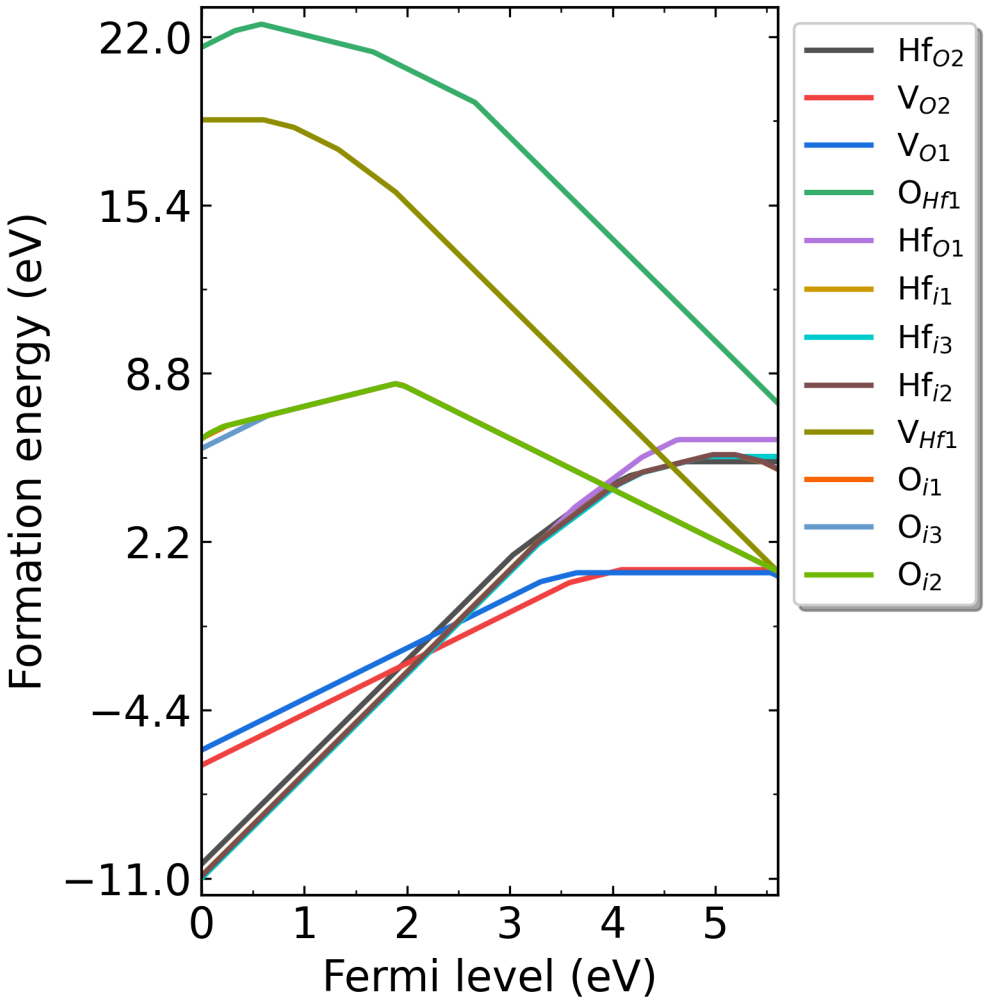
Meanwhile, the image of transition energy levels for each defect is also output, as shown in below:

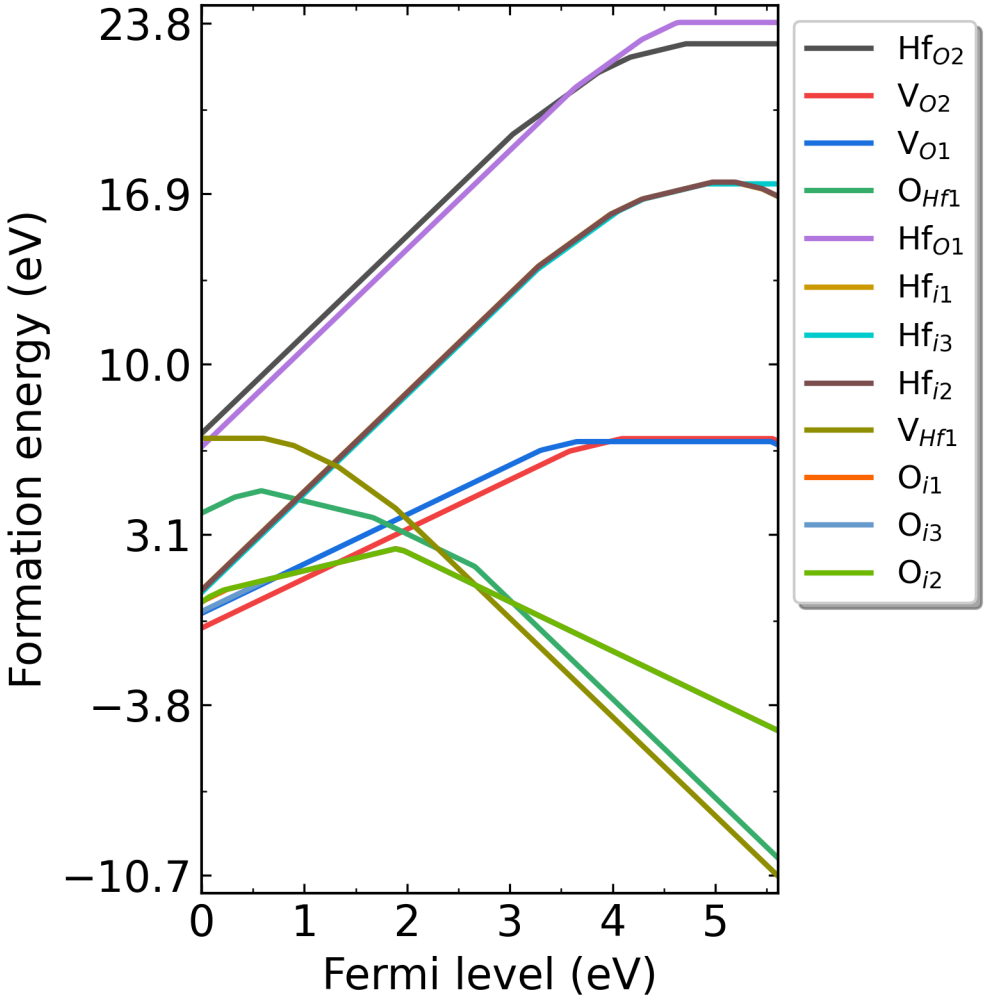
Fig: The charge-state transition energy levels of intrinsic defects in HfO2.

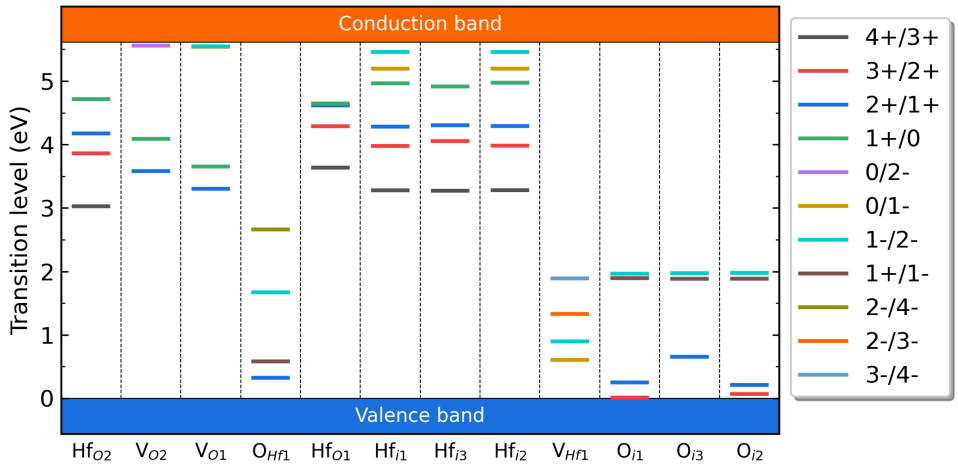
5.2.4 5.2.4 DDC -- defect density and Fermi level calculations

5.2.4.1 Run DDC module

After finishing the DEC module, return to the directory HfO2 and use the command *dasp 4* to execute the DDC module. No additional operation is required while waiting for the program to complete.


22.0
15.4
8.8
2.2
−4.4
−11.0
Formation energy (eV)
0
1
2
3
4
5
Fermi level (eV)
Hf_{O2}
V_{O2}
V_{O1}
O_{Hf1}
Hf_{O1}
Hf_{i1}
Hf_{i3}
Hf_{i2}
V_{Hf1}
O_{i1}
O_{i3}
O_{i2}


23.8
16.9
10.0
3.1
−3.8
−10.7
Formation energy (eV)
0
1
2
3
4
5
Fermi level (eV)
Hf_{O2}
V_{O2}
V_{O1}
O_{Hf1}
Hf_{O1}
Hf_{i1}
Hf_{i3}
Hf_{i2}
V_{Hf1}
O_{i1}
O_{i3}
O_{i2}


Conduction band
Valence band
Transition level (eV)
0
1
2
3
4
5
Hf_{O2}
V_{O2}
V_{O1}
O_{Hf1}
Hf_{O1}
Hf_{i1}
Hf_{i3}
Hf_{i2}
V_{Hf1}
O_{i1}
O_{i3}
O_{i2}
4+/3+
3+/2+
2+/1+
1+/0
0/2-
0/1-
1-/2-
1+/1-
2-/4-
2-/3-
3-/4-

5.2.4.2 Workflow of DDC module

Summarize the defect-related data:

Firstly, DDC module will judge which defects have been finished calculating based on the output of DEC module, and take all these defects into consideration in the DDC calculation. Then, DDC will search for the output information about each defect as formation energy, transition energy levels, and degeneracy factor. Finally, summarize all the data and write in the file `DefectParams.txt` .

`4ddc.out` is the log file of DDC module:

```
############ Collecting information from DEC ############
Read defect types from DEC calculation successfully.
Defects considered in DDC calculation: ['Hf_O2', 'V_O2', 'V_O1', 'O_Hf1', 'Hf_O1', 'Hf_i-
↪1', 'Hf_i-3', 'Hf_i-2', 'V_Hf1', 'O_i-1', 'O_i-3', 'O_i-2']
Chemical potentials change from p1 to p2.
Calculate gq for defect in each charge state.
Calculate Nsites for Hf_O2: 5.708701e+22 cm^-3.
Calculate Nsites for V_O2: 5.708701e+22 cm^-3.
Calculate Nsites for V_O1: 5.708701e+22 cm^-3.
Calculate Nsites for O_Hf1: 2.854350e+22 cm^-3.
Calculate Nsites for Hf_O1: 5.708701e+22 cm^-3.
Calculate Nsites for Hf_i-1: 2.854350e+22 cm^-3.
Calculate Nsites for Hf_i-3: 2.854350e+22 cm^-3.
Calculate Nsites for Hf_i-2: 2.854350e+22 cm^-3.
Calculate Nsites for V_Hf1: 2.854350e+22 cm^-3.
Calculate Nsites for O_i-1: 2.854350e+22 cm^-3.
Calculate Nsites for O_i-3: 2.854350e+22 cm^-3.
Calculate Nsites for O_i-2: 2.854350e+22 cm^-3.
############ Collecting information from DEC ############
```

Below is the file `DefectParams.txt` :

```
800 300
2.950000 2.990000
5.611337
Hf_O2 5.708701e+22 1 4.714 2 4.444 1 4.25 2 3.944 1 x x x x x x x 5.340000 22.964000
V_O2 5.708701e+22 1 4.089 2 3.835 1 x x x x 5.621 2 5.558 1 x x x x 1.106000 6.981000
V_O1 5.708701e+22 1 3.652 2 3.477 1 x x x x 5.539 2 5.542 1 x x x x 0.986000 6.861000
O_Hf1 2.854350e+22 1 0.79 2 0.557 1 x x x x 0.374 2 1.022 1 1.689 2 1.841 1 22.704000 5.
↪080000
Hf_O1 5.708701e+22 1 4.646 2 4.631 1 4.517 2 4.296 1 x x x x x x x x 6.205000 23.830000
Hf_i-1 2.854350e+22 1 4.964 2 4.622 1 4.407 2 4.125 1 5.194 2 5.326 1 x x x x 5.601000␣
↪17.351000
Hf_i-3 2.854350e+22 1 4.914 2 4.608 1 4.423 2 4.135 1 x x x x x x x x 5.544000 17.294000
Hf_i-2 2.854350e+22 1 4.974 2 4.632 1 4.415 2 4.131 1 5.194 2 5.326 1 x x x x 5.614000␣
↪17.364000
```

(continues on next page)

(continued from previous page)

```
V_Hf1 2.854350e+22 1 x x x x x x x x 0.606 2 0.751 1 0.944 2 1.18 1 18.743000 6.994000
O_i-1 2.854350e+22 1 2.09 2 1.17 1 0.784 2 x x 1.697 2 1.831 1 x x x x 8.596000 2.722000
O_i-3 2.854350e+22 1 2.093 2 1.374 1 0.814 2 x x 1.675 2 1.824 1 x x x x 8.606000 2.
↪731000
O_i-2 2.854350e+22 1 2.095 2 1.153 1 0.791 2 x x 1.674 2 1.825 1 x x x x 8.606000 2.
↪731000
```

Self-consistent calculation under growth temperature:

The DDC module will calculate the defect/dopant and carrier densities at the temperature T=800 K, and obtain the Fermi level by self-consistently under the charge neutralization condition.

Self-consistent calculation under working (measuring) temperature:

The DDC module will recalculate the defect/dopant and carrier densities at the temperature T=300 K, and re-obtain the Fermi level by self-consistently under the charge neutralization condition.

Output defect density:

DDC module will output three files in the directory CdTe/ddc: `Fermi.dat` , `Carrier.dat` , `Defect_charge.dat` , which can be plotted using Origin. In addition, DDC also can automatically generate image file `density.png` based on these three files, as follows:

Fig: The Fermi level, electron and hole carrier densities, and defect densities in HfO2 as functions of the chemical potentials (from Hf-rich to O-rich) with a growth temperature is 300 K.

Fig: The Fermi level, electron and hole carrier densities, and defect densities in HfO2 as functions of the chemical potentials (from Hf-rich to O-rich) with a growth temperature is 550 K.

Fig: The Fermi level, electron and hole carrier densities, and defect densities in HfO2 as functions of the chemical potentials (from Hf-rich to O-rich) with a growth temperature is 800 K.

5.3 5.3 The Defect calculations for H-doped Ga2O3

Ga2O3 is a transparent conductive oxide material with a natural n-type conductivity and high optical transparency. It is used for various application fields, such as flat panel displays, touch panels and displays, top surface electrodes of solar cells, and solid-state light emitters.

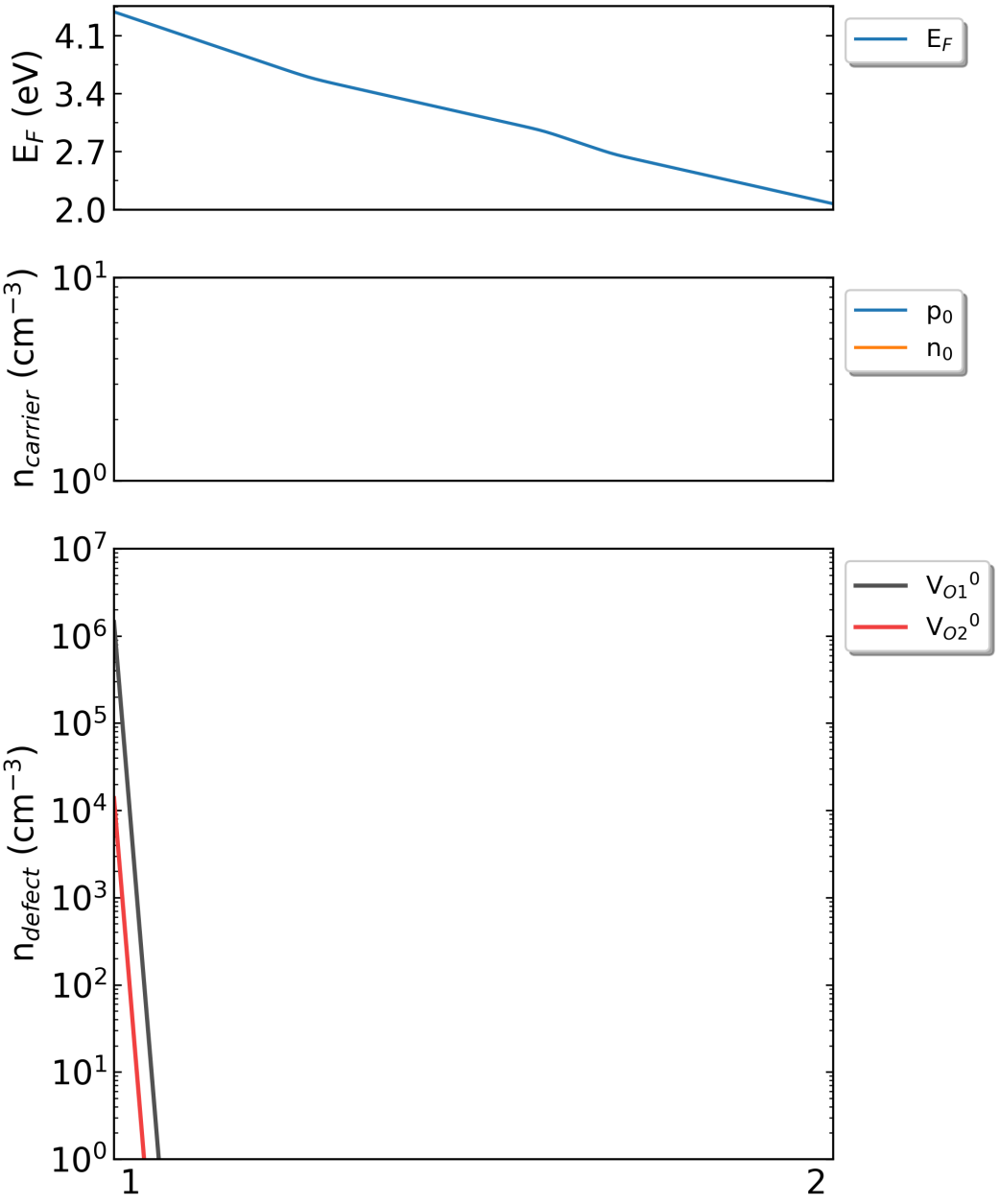
Ga2O3 thin films and crystals show n-type conductivity without external doping, and the origin is controversial. Intrinsic defects and dopants both are possible inducements.

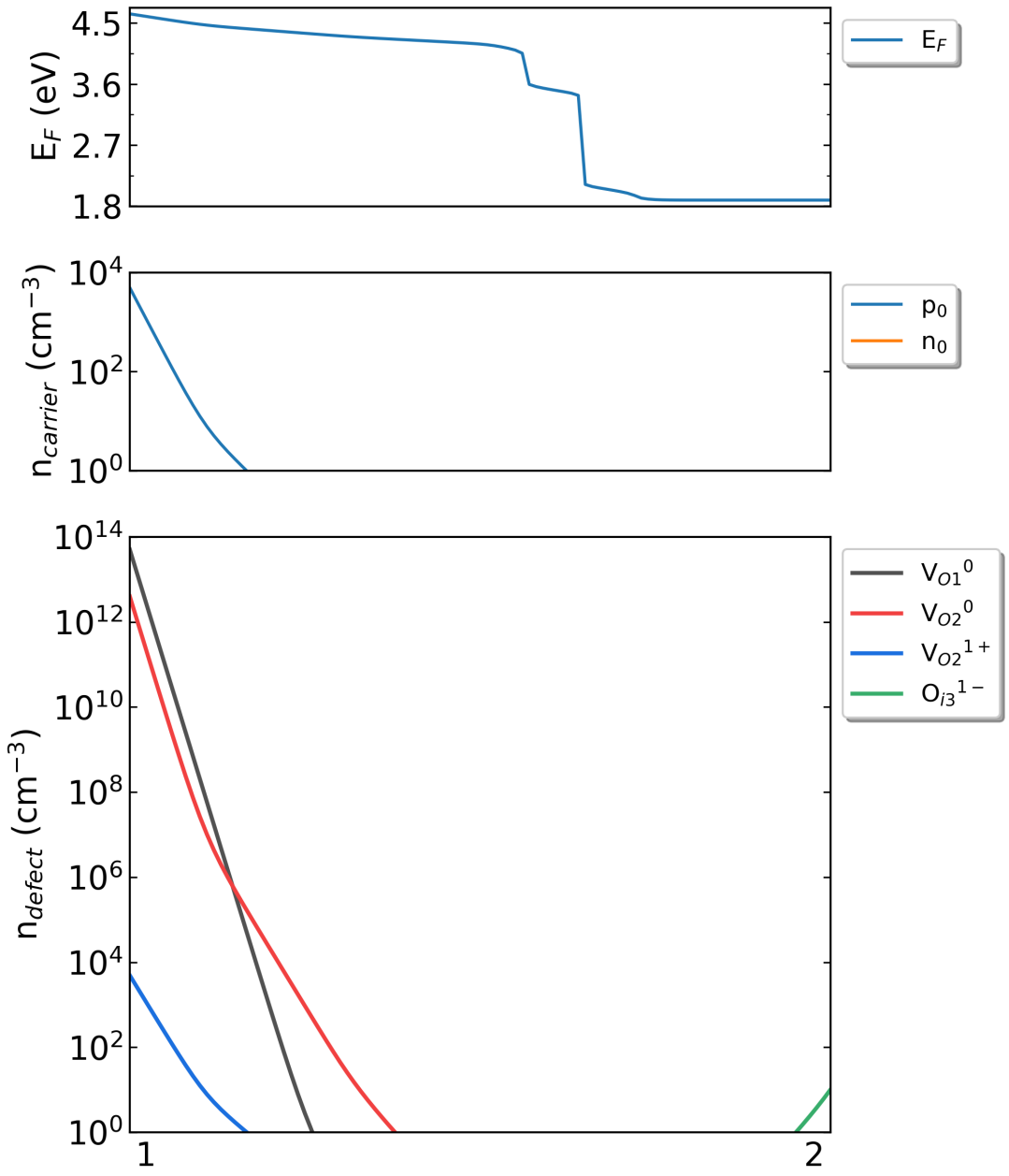
With the help of DASP software, the properties of the point defects and dopants can be systematically studied, and the mechanism of defects and dopants formation also can be explored to reveal the relationship between them and conductivity.

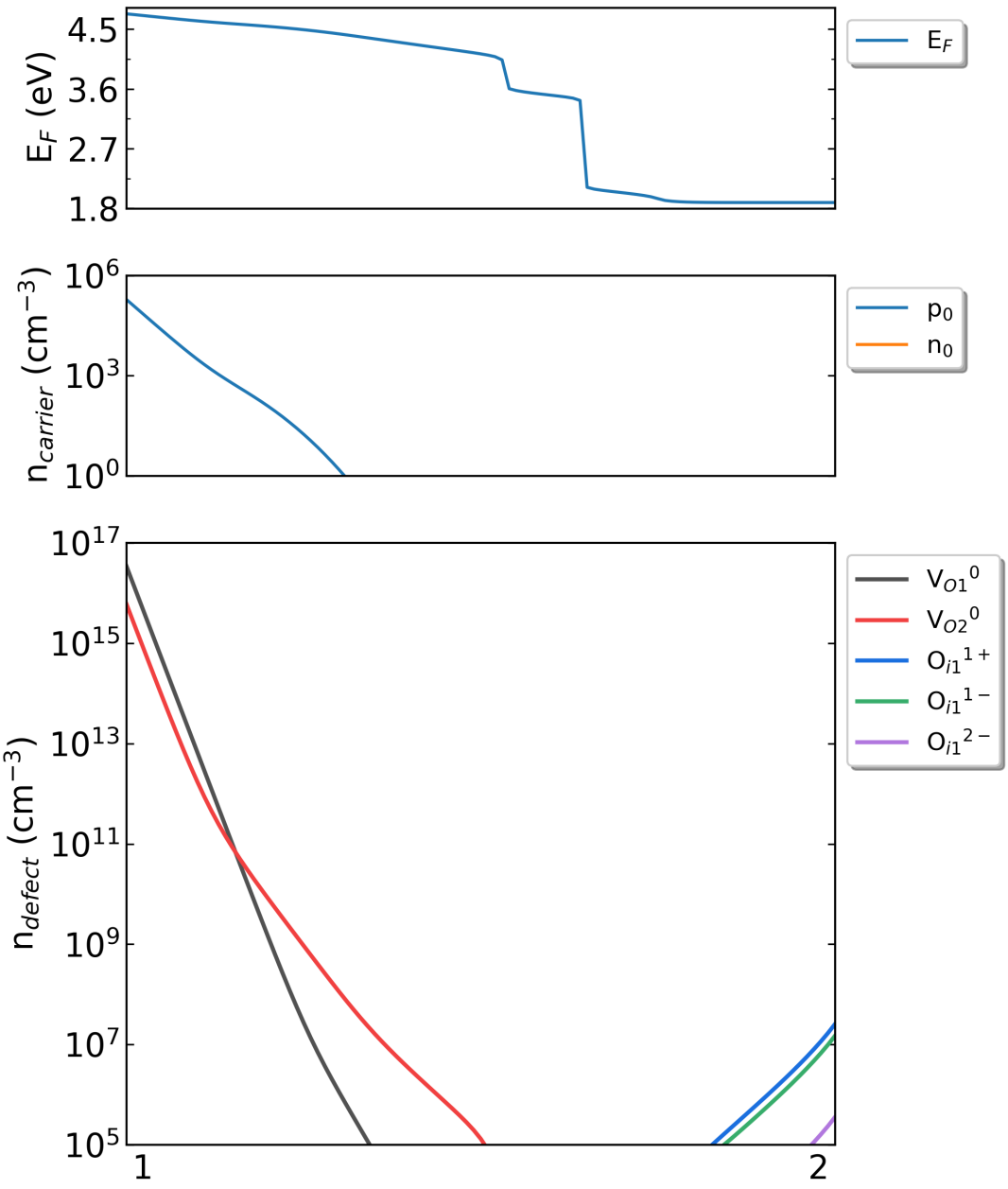
5.3.1 5.3.1 PREPARE --- Prepares for calculation

5.3.1.1 POSCAR and dasp.in

Find the `POSCAR` for Ga2O3 from the `Materials Project` database, as follows:


4.1
3.4
2.7
2.0
E_F (eV)
E_F
10^1
10^0
$n_{carrier}$ (cm^{-3})
p_0
n_0
10^7
10^6
10^5
10^4
10^3
10^2
10^1
10^0
n_{defect} (cm^{-3})
$V_{O1}{}^0$
$V_{O2}{}^0$
1
2


4.5
3.6
2.7
1.8
E_F (eV)
E_F
10^4
10^2
10^0
$n_{carrier}$ (cm^{-3})
p_0
n_0
10^{14}
10^{12}
10^{10}
10^8
10^6
10^4
10^2
10^0
n_{defect} (cm^{-3})
V_{O1}^{0}
V_{O2}^{0}
V_{O2}^{1+}
O_{i3}^{1-}
1
2


4.5
3.6
2.7
1.8
E_F (eV)
E_F
10^6
10^3
10^0
$n_{carrier}$ (cm^{-3})
p_0
n_0
10^{17}
10^{15}
10^{13}
10^{11}
10^9
10^7
10^5
n_{defect} (cm^{-3})
1
2
V_{O1}^{0}
V_{O2}^{0}
O_{i1}^{1+}
O_{i1}^{1-}
O_{i1}^{2-}

```
Ga8 O12
1.0
          12.2299995422         0.0000000000         0.0000000000
           0.0000000000         3.0399999619         0.0000000000
          -1.3736609922         0.0000000000         5.6349851545
   Ga    O
    8   12
Direct
         0.158409998          0.500000000          0.314081997
         0.341589987          0.000000000          0.685917974
         0.089878000          0.000000000          0.794761002
         0.410122007          0.500000000          0.205238998
         0.658410013          0.000000000          0.314081997
         0.841589987          0.500000000          0.685917974
         0.589878023          0.500000000          0.794761002
         0.910121977          0.000000000          0.205238998
         0.495896995          0.000000000          0.256491005
         0.004103000          0.500000000          0.743508995
         0.173598006          0.000000000          0.564293981
         0.326402009          0.500000000          0.435705990
         0.336497009          0.500000000          0.891047001
         0.163503006          0.000000000          0.108952999
         0.995896995          0.500000000          0.256491005
         0.504103005          0.000000000          0.743508995
         0.673597991          0.500000000          0.564293981
         0.826402009          0.000000000          0.435705990
         0.836497009          0.000000000          0.891047001
         0.663502991          0.500000000          0.108952999
```

Taking it into the crystal visualization software, the structure can be seen as below:

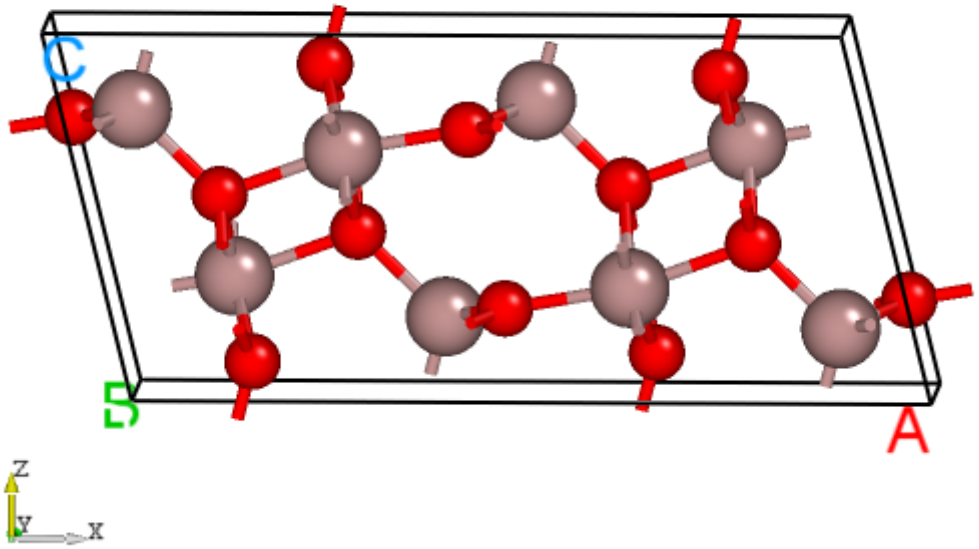
The structure of Ga2O3.

Next, use VASP to optimize its lattice parameters or modify the lattice to match the experimental measurements. Users need to do it manually.

Write the required parameters in `dasp.in` :

```
############### Job Scheduling ###############
cluster = SLURM     # (job scheduling system)
node_number = 4          # (number of node)
core_per_node = 52       # (core per node)
queue = batch            # (name of queue/partition)
max_time = 24:00:00         # (maximum time for a single DFT calculation)
vasp_path_dec = /opt/vasp.5.4.4/bin/vasp_gam   #  (path of VASP)
vasp_path_tsc = /opt/vasp.5.4.4/bin/vasp_std
job_name = submit_job    # (name of script)
```

(continues on next page)



(continued from previous page)

```
potcar_path = /opt/POT/potpaw_PBE    # (path of pseudopotentials)
max_job = 5

############### TSC Module ###############
database_api = ******************** # (str-list type)

############### DEC Module ###############
level = 2   # (level=1: PBE+PBE; level=2: PBE+HSE; level=3: HSE+HSE)
min_atom = 200
max_atom = 250
intrinsic = F   # (default: T)
doping = T   # (default: F)
impurity = H
correction = FNV   # (default: none)
epsilon = 10.8
Eg_real = 4.9   # (experimental band gap)

############### DDC Module ###############
ddc_temperature = 1124 300
ddc_mass = 0.23 2.90
```

Next, all the parameters listed in `dasp.in` will be described:

```
cluster = SLURM
# The system of the used cluster is SLURM.
```

```
node_number = 4
# 4 nodes are used for each calculation.
```

```
core_per_node = 52
# 52 cores are used for each node, so 4*52=208 cores are used in total for each␣
↪calculation.
```

```
queue = batch
# The queue named "batch" is used to carry out calculations. Therefore, users need to␣
↪make sure the queue name, nodes, and cores of clusters before configuring dasp.in.
```

```
max_time = 24:00:00        # (maximum time for a single DFT calculation)
# The maximum time allowed is 24 hours for a single DFT calculation and can be set␣
↪arbitrarily.
```

```
vasp_path_dec = /opt/vasp.5.4.4/bin/vasp_gam   #  (path of VASP)
vasp_path_tsc = /opt/vasp.5.4.4/bin/vasp_std
# The VASP_std version is used for TSC calculations, and the VASP_gam version is used␣
↪for DEC calculations.
```

```
job_name = submit_job    # (name of script)
# The submission script, named " submit_job" and can be set arbitrarily.
```

```
potcar_path = /opt/POT/potpaw_PBE    # (path of pseudopotentials)
# path of pseudopotentials
```

```
max_job = 5
# the allowed maximum number of jobs at the same time
```

```
database_api = ******************** # (str-list type)
# using to visit the Materials Project database
```

```
level = 2   # (level=1: PBE+PBE; level=2: PBE+HSE; level=3: HSE+HSE)
# using GGA-PBE for structural relaxation and HSE to calculate the total energy
```

```
min_atom = 200
max_atom = 250
# The number of atoms within the generated supercell that we want is between 200 and 250,
↪ and as far as possible to make a=b=c and a b c.
```

```
intrinsic = F   # (default: T)
# Don' t generate intrinsic defects.
```

```
doping = T   # (default: F)
# Generate dopants.
```

```
impurity = H
# The doping element is H, and generate defects H_Ga, H_O, and H_i.
```

```
correction = FNV   # (default: none)
# The corrections for charged defect adopt FNV correction.
```

```
epsilon = 10.8
# The dielectric constant of Ga2O3 is 10.8.
```

```
Eg_real = 4.9   # (experimental band gap)
# The experimental band gap of Ga2O3 is about 4.9 eV, DASP will adjust AEXX in INCAR to␣
↪make the band gap of the supercell without defect equal to 4.9 eV.
```

```
ddc_temperature = 1000 300
# the growth temperature set to 1000 K and the working temperature set to 300 K.
```

```
ddc_mass = 0.23 4.21
# electron effective mass set to 0.23 and hole effective mass set to 4.21.
```

5.3.1.2 Use DASP to generate the required input files

Create a new directory doping-Ga2O3, then prepare the files, `POSCAR` and `dasp.in` , mentioned above in the directory ./doping-Ga2O3/. Next, execute `dasp 1` to start PREPARE module and no additional operation is needed thereafter. DASP will output file `1prepare.out` to record the running log of the program.

5.3.1.3 Workflow of PREPARE module

Generate supercell:

Firstly, the program will automatically find the optimal supercell (as far as possible to make a=b=c and a b c) based on the parameters min_atom=200 and max_atom=250 and output POSCAR for the supercell. The following are the structure messages of Ga2O3 supercell `POSCAR_nearlycube` :

```
Cubic_cell
1.0
18.9887859181 0.0000000000 0.0000000000
-1.4600617135 9.0023673390 0.0000000000
0.7906182108 0.1282275357 14.7068652328
Ga O
96 144
Direct
0.1256845017 0.0418948339 0.5327255075
0.3743154982 0.2914384994 0.4672744924
0.2212487535 0.2404162511 0.3686292617
0.2787512464 0.0929170821 0.6313707382
0.8756845017 0.1252281672 0.2827255075
```

(continues on next page)

(continued from previous page)

```
0.1243154982 0.0414384994 0.2172744924

...
```

Taking it into the crystal visualization software, the supercell can be seen as below:

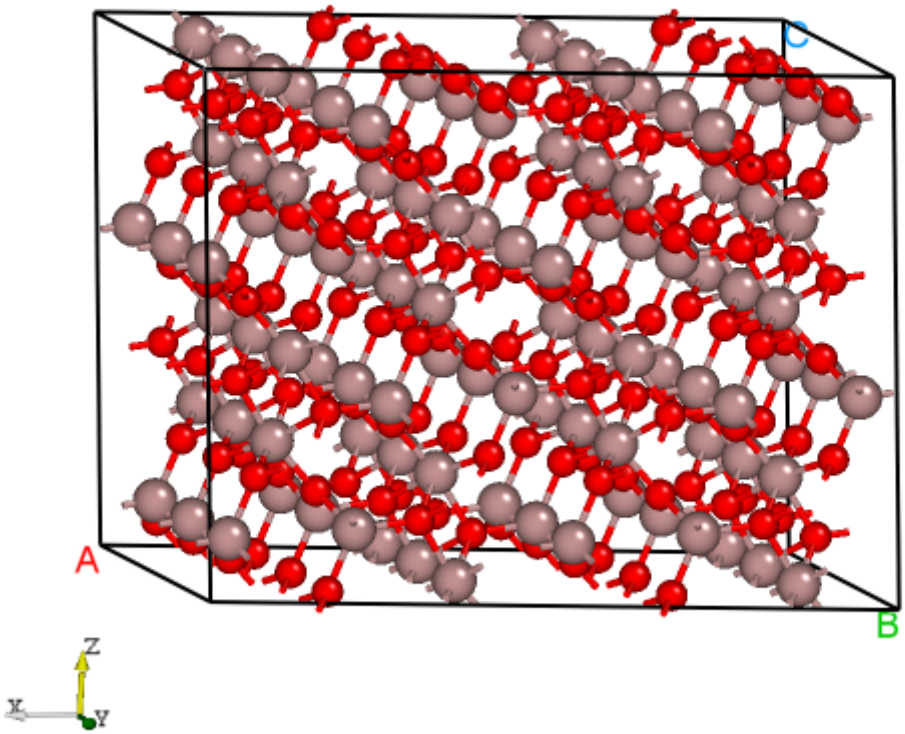


Fig: The structure of Ga2O3 supercell generated by DASP.

Madelung constant calculation:

Secondly, according to the generated supercell file, the program will execute Madelung constant calculation which describes the Coulomb interaction between charged defect and periodic image charge. (use for Lany-Zunger correction)

After finishing the above two steps calculation, the output of `1prepare.out` is as follows:

```
############ Prepare Files module start ############

Read the structure file POSCAR you provided
Get the refined cell POSCAR_refined from POSCAR
Generate the nearlycube cell POSCAR_nearlycube from POSCAR
Generate job script through dasp.in parameters
Generate single-point KPOINTS
Generate pseudopotential file POTCAR through potcar_dir you set
Generate commonly used vasp input file INCAR
Start the madelung constant calculation
Generate the madelung calculation directory
Generate madelung calculation POSCAR
Generate madelung calculation POTCAR
Generate madelung calculation INCAR
Generate madelung calculation KPOINTS
Generate madelung calculation job script
Job 503.host5 submitted: /data2/home/chensy/zzn/doping-Ga2O3/dec/madelung/static
Succeed job 503.host5: /data2/home/chensy/zzn/doping-Ga2O3/dec/madelung/static
The madelung constant calculation completed
The madelung constant = 2.411
```

HSE exchange proportion calculation:

According to the generated supercell file, the program will perform HSE static calculations with AEXX=0.25 and AEXX=0.3 respectively to determine the value of AEXX which can make the obtained band gap match *Eg_real = 4.9* based on the slope. If the calculated band gap with AEXX=0.25 or AEXX=0.3 is consistent with the set, the subsequent calculation for AEXX will not be performed. Therefore, after the calculation is completed, the contents in the directory doping-Ga2O3/dec/AEXX/ are as follows:

```
cd ./dec/AEXX
ls
0.25  0.3  0.3292780889291405  AEXX.list
```

It indicates that the band gap of the Ga2O3 supercell is 4.9 eV when AEXX=0.33 (two decimal places), and write this parameter into INCAR. Meanwhile, the log can be seen from `1prepare.out` as follows:

```
Start the HSE parameter AEXX calculation
Job 507.host5 submitted: /data2/home/chensy/zzn/doping-Ga2O3/dec/AEXX/0.25/static
Succeed job 507.host5: /data2/home/chensy/zzn/doping-Ga2O3/dec/AEXX/0.25/static
Job 508.host5 submitted: /data2/home/chensy/zzn/doping-Ga2O3/dec/AEXX/0.3/static
Succeed job 508.host5: /data2/home/chensy/zzn/doping-Ga2O3/dec/AEXX/0.3/static
Job 509.host5 submitted: /data2/home/chensy/zzn/doping-Ga2O3/dec/AEXX/0.3292780889291405/
↪static
Succeed job 509.host5: /data2/home/chensy/zzn/doping-Ga2O3/dec/AEXX/0.3292780889291405/
↪static
The HSE parameter AEXX calculation completed
```

(continues on next page)

(continued from previous page)

```
The HSE parameter AEXX = 0.33
level = 2: Generate PBE relax vasp input file INCAR-relax
level = 2: Generate HSE static vasp input file INCAR-static
```

Optimize the ionic position of the host supercell:

The last step in PREPARE module is to optimize the ionic position of the host supercell according to level=2 (i.e. PBE relax). The optimized file is `POSCAR_final` in the directory doping-Ga2O3/dec/relax. At the same time, the sign of the end of DASP operation can be seen in `1prepare.out` , and it also tells us that we need to do the TSC module calculation in the next step.

```
Start the POSCAR_nearlycube relax calculation
Generate the POSCAR_nearlycube relax directory
Job 510.host5 submitted: /data2/home/chensy/zzn/doping-Ga2O3/dec/relax
Succeed job 510.host5: /data2/home/chensy/zzn/doping-Ga2O3/dec/relax
The POSCAR_nearlycube relax calculation completed
Get the final structure POSCAR_final

############ Prepare Files module end ############

DASP-PREPARE finished, please run DASP-TSC next
```

5.3.2 5.3.2 TSC -- thermodynamic stability and chemical potential calculations

5.3.2.1 Run TSC module

The directory doping-Ga2O3/dec will be created when using the command *dasp 1* to execute PREPARE module, and generate file `1prepare.out` in this directory. After finishing the program, there has the corresponding completion flag in `1prepare.out` . Then, enter the directory doping-Ga2O3/dec and confirm that the parameters in `INCAR-relax` and `INCAR-static` are feasible. (Users can modify INCAR, and DASP will make subsequent calculations based on the INCAR in this directory.)

Once confirm that the PREPARE module is finished, return to the directory doping-Ga2O3 and use the command *dasp 2* to execute the TSC module. Similarly, the TSC module will create a directory named tsc under the directory doping-Ga2O3, which contains the output of the TSC program, including every calculation directory and the running log file `2tsc.out` . No additional operation is required while waiting for the program to complete.

5.3.2.2 Workflow of TSC module

The total energy calculation of the host structure (the parameters are consistent with MP database):

TSC module will use the same input parameters (INCAR, KPOINTS, POTCAR) with the `Materials Project` database to perform structural relaxation and static calculation on the primitive cells given by the user. Therefore, the calculated total energy is comparable to that of the MP database. This step is to obtain the **key hetero-phases** that limit the stability of Ga2O3. In the directory, we can see:

```
cd tsc
cd Ga2O3/
ls
relaxation1  relaxation2  static
```

The running log also can be seen from the doping-Ga2O3/tsc/2tsc.out, that is, the steps such as generating input files, relaxation1, relaxation2, static and data extraction.

The judgement of key hetero-phases compounds:

The TSC module will search for all the secondary compounds that compete with Ga2O3 in the MP database. And compare the total energy of Ga2O3 calculated in the previous step with that of the hetero-phases extracted from the database to confirm Ga2O3 is **thermodynamically stable**. Subsequently, the program will automatically download the key hetero-phase compounds that can limit the thermodynamic stability of H-doped Ga2O3. Only H and GaHO2 are considered in this case. The relevant information can be seen in `2tsc.out` :

```
...
analysing the thermodynamic stability of Ga2O3.
The stability of Ga2O3 is: True.
key phases of Ga2O3 are: Ga O2 .
key phases of H doped Ga2O3 are: H2 GaHO2 .
analysing of Ga2O3 is done.
sub-module of tsc: 'auto thermodynamic calculation' ends successfully.
...
```

The total energy calculation of the host and hetero-phase compounds:

After the key hetero-phase compounds are confirmed, TSC will calculate the total energy of Ga2O3, GaHO2, and H2 by using the parameter (AEXX) obtained from PREPARE module. `2tsc.out` is as follows:

```
...
Job 520.host5 submitted: /data2/home/chensy/zzn/doping-Ga2O3/tsc/GaHO2/static_recalc
Job 521.host5 submitted: /data2/home/chensy/zzn/doping-Ga2O3/tsc/H2/static_recalc
Succeed job  520.host5: /data2/home/chensy/zzn/doping-Ga2O3/tsc/GaHO2/static_recalc
Succeed job  521.host5: /data2/home/chensy/zzn/doping-Ga2O3/tsc/H2/static_recalc
...
```

The chemical potential calculation:

Calculating the formation energy and stable chemical potential region of H-doped Ga2O3 based on the calculated total energy. As Ga2O3 is a binary compound, TSC module will give the endpoint of two chemical potentials, i.e. Ga-rich and O-rich, and write them into `dasp.in` :

```
# The orders are consistent with the order of elements in POSCAR, i.e. the first column␣
↪is Ga, the second column is O, and the third is the dopant H.
E_pure = -4.1294 -9.4157 1.0503
p1 = 0.0 -3.72 -4.9467
p2 = -5.5801 0.0 -6.8066
```

The output after the program is completed can be seen in `2tsc.out` :

```
dir '2d-figures','3d-figures','ori_data_MP' ready.  analysing the thermodynamic␣
↪stability of Ga2O3.
The stability of Ga2O3 is: True.
key phases of Ga2O3 are: Ga O2 .
key phases of H doped Ga2O3 are: H2 GaHO2 .
analysing of Ga2O3 is done.
sub-module of tsc: 'auto thermodynamic calculation' ends successfully.
--------------------------
DASP-TSC finished
```

For the ternary and multinary compounds, TSC module will output the image of the stable region and the chemical potential at the endpoint of the stable region.

5.3.3 5.3.3 DEC -- the calculations of defect formation energy and transition energy level

5.3.3.1 Run DEC module

The directory doping-Ga2O3/tsc will be created when using the command *dasp 2* to execute TSC module, and generate file `2tsc.out` in this directory. After finishing the TSC module, there has the corresponding completion flag in `2tsc.out` . Then, open the file `dasp.in` under the directory doping-Ga2O3/dasp.in to confirm the chemical potential already has been written.

Once confirm that the TSC module is finished, return to the directory doping-Ga2O3 and use the command *dasp 3* to execute the DEC module. DEC will output relevant files in the generated directory dec in the first step, including the defect structures, directories, and the log file `3dec.out` . No additional operation is required while waiting for the program to complete.

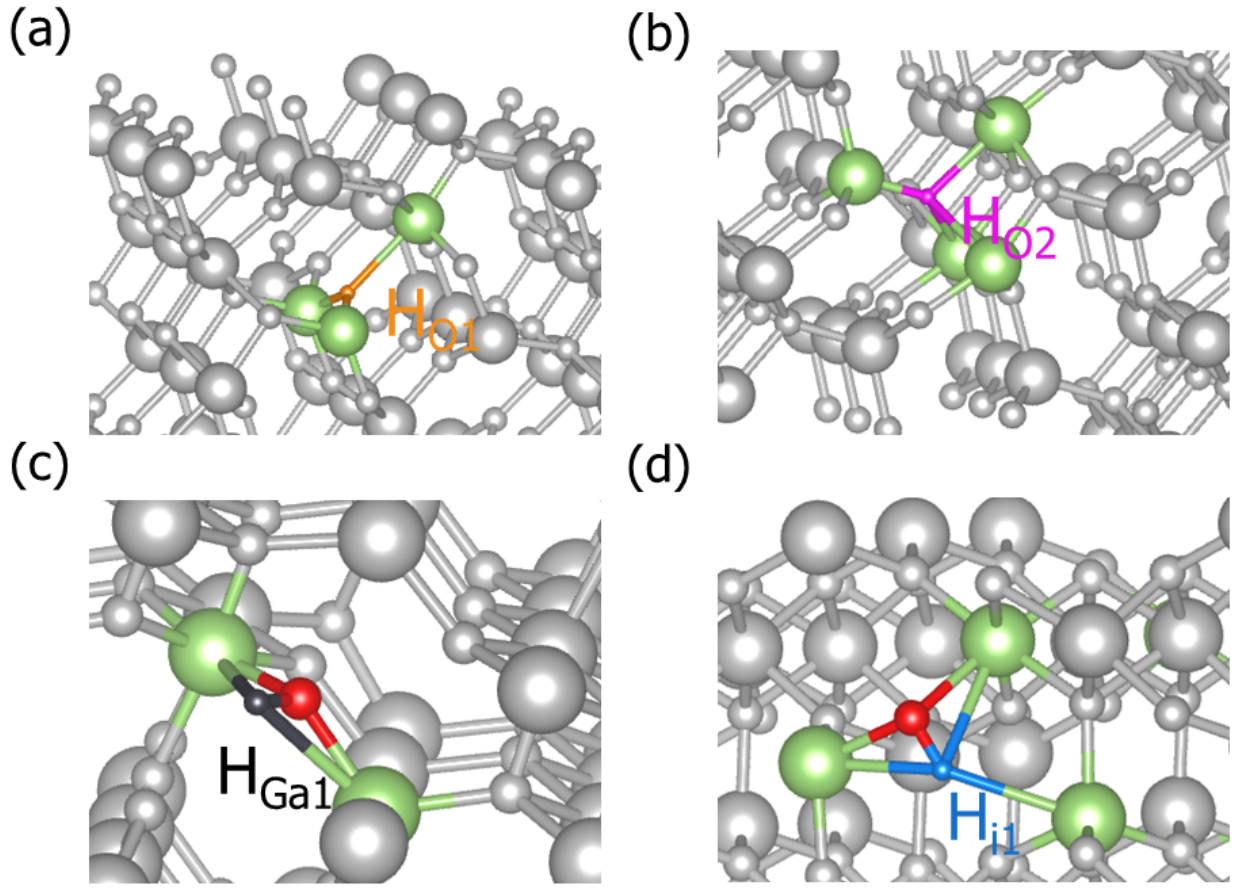
5.3.3.2 Workflow of DEC module

Generate defect structure:

Based on the parameters doping = T and impurity = H in `dasp.in` , DEC will generate the H-doped defects for Ga2O3, i.e. create the calculation directory doping-Ga2O3/dec/Doping-H, in which the structures and directories of substitutions H_Ga and H_O, and interstitial defect H_i. According to the crystal symmetry analysis, there have two inequivalent sites for Ga atom, and three inequivalent sites for O atom. Therefore, there will generate two different configurations for H_Ga and three configurations for H_O, while the number of configurations for H_i is depended on the input parameter set by users.

```
cd dec/Doping_H/
ls
Doping_H.list  H_Ga1  H_Ga2  H_i  H_O1  H_O2  H_O3  host
```

Take some defect structures to see in the visualization software, as shown in below:



Part of defect structures of H-doped Ga2O3 generated by DASP.

At the same time, the output of DEC module can be seen in `3dec.out` as follows:

```
############ Neutral Defect module start #############
Make doping defect directory Doping_H
Generate host directory in Doping_H
Start generating neutral doping_H antisite defect
Generate neutral defect at: H_Ga1/initial_structure/q0
Generate neutral defect at: H_Ga2/initial_structure/q0
Generate neutral defect at: H_O1/initial_structure/q0
Generate neutral defect at: H_O2/initial_structure/q0
```

(continues on next page)

(continued from previous page)

```
Generate neutral defect at: H_O3/initial_structure/q0
Neutral doping_H substitution defect generation completed
Start generating neutral doping_H interstitial defect
Generate neutral defect at: H_i/random1/initial_structure/q0
Generate neutral defect at: H_i/random2/initial_structure/q0
Generate neutral defect at: H_i/random3/initial_structure/q0
Generate neutral defect at: H_i/random4/initial_structure/q0
Generate neutral defect at: H_i/random5/initial_structure/q0
Generate neutral defect at: H_i/random6/initial_structure/q0
Neutral doping_H interstitial defect generation completed

############ Neutral Defect module end ############
```

It can be seen that the DEC module only creates the directories for all the neutral defects at this time.

Submit jobs for all defects with q=0:

After the structures and directories of neutral defects are generated, DEC module will call VASP to perform structural relaxation with PBE and total energy calculation with HSE (corresponds to the parameter level=2 in `dasp.in`), this step may need a long time. Users can check the file `3dec.out` at any time. The messages in `3dec.out` are as follows:

```
Job 598.host5 submitted: /data2/home/chensy/zzn/doping-Ga2O3/H_Ga1/initial_structure/q0
Job 600.host5 submitted: /data2/home/chensy/zzn/doping-Ga2O3/H_Ga2/initial_structure/q0
Job 602.host5 submitted: /data2/home/chensy/zzn/doping-Ga2O3/H_O1/initial_structure/q0
Job 604.host5 submitted: /data2/home/chensy/zzn/doping-Ga2O3/H_O2/initial_structure/q0
Job 606.host5 submitted: /data2/home/chensy/zzn/doping-Ga2O3/H_O3/initial_structure/q0

...

Succeed job 602.host5: /data2/home/chensy/zzn/doping-Ga2O3/H_O1/initial_structure/q0
Succeed job 600.host5: /data2/home/chensy/zzn/doping-Ga2O3/H_Ga2/initial_structure/q0
Succeed job 598.host5: /data2/home/chensy/zzn/doping-Ga2O3/H_Ga1/initial_structure/q0
Succeed job 604.host5: /data2/home/chensy/zzn/doping-Ga2O3/H_O2/initial_structure/q0
Succeed job 606.host5: /data2/home/chensy/zzn/doping-Ga2O3/H_O3/initial_structure/q0

...
```

Generate calculation directories for the charged defects:

After finishing the calculations of all the neutral defects (except for interstitial defects with high energy), the program will judge the charge states of each defect and generate the corresponding directories and files for the charged defects based on the results of the neutral defects. A prompt will be given for those defects with calculation errors (undo, failed, and not converged) or without subsequent calculation (skip). The relevant information in `3dec.out` is as follows:

```
############ Ionized Defect module start ############
Start generating ionized defects
Ionized defect path: /data2/home/chensy/zzn/doping-Ga2O3/H_O1/initial_structure/q+1
Ionized defect path: /data2/home/chensy/zzn/doping-Ga2O3/H_O1/initial_structure/q+2
Ionized defect path: /data2/home/chensy/zzn/doping-Ga2O3/H_O1/initial_structure/q+3
Ionized defect path: /data2/home/chensy/zzn/doping-Ga2O3/H_O1/initial_structure/q+4
Ionized defects generation completed
Start generating ionized defects
Ionized defect path: /data2/home/chensy/zzn/doping-Ga2O3/H_O2/initial_structure/q+1
Ionized defect path: /data2/home/chensy/zzn/doping-Ga2O3/H_O2/initial_structure/q+2
Ionized defect path: /data2/home/chensy/zzn/doping-Ga2O3/H_O2/initial_structure/q+3
Ionized defect path: /data2/home/chensy/zzn/doping-Ga2O3/H_O2/initial_structure/q+4
Ionized defects generation completed
The static calculation of /data2/home/chensy/zzn/doping-Ga2O3/H_i/random3/initial_
↪structure/q0/static is skipped, skip ionized defect generation

...
```

Submit jobs for the defects with q 0:

After the structures and directories of the charged defects are generated, DEC module will call VASP to perform structural relaxation with PBE and total energy calculation with HSE (corresponds to the parameter level=2 in `dasp.in`). The waiting time needed in this step will be longer than that in 3.2.2. The messages in `3dec.out` are as follows:

```
############ AutoRun - Ionized Defect module start ############
Job 659.host5 submitted: /data2/home/chensy/zzn/doping-Ga2O3/H_O1/initial_structure/q+2
Job 661.host5 submitted: /data2/home/chensy/zzn/doping-Ga2O3/H_O1/initial_structure/q+1
Job 663.host5 submitted: /data2/home/chensy/zzn/doping-Ga2O3/H_O1/initial_structure/q+3
Job 665.host5 submitted: /data2/home/chensy/zzn/doping-Ga2O3/H_O1/initial_structure/q-1
Job 667.host5 submitted: /data2/home/chensy/zzn/doping-Ga2O3/H_O2/initial_structure/q+2

...

Succeed job 659.host5: /data2/home/chensy/zzn/doping-Ga2O3/H_O1/initial_structure/q+2
Succeed job 661.host5: /data2/home/chensy/zzn/doping-Ga2O3/H_O1/initial_structure/q+1
Succeed job 663.host5: /data2/home/chensy/zzn/doping-Ga2O3/H_O1/initial_structure/q+3
Succeed job 665.host5: /data2/home/chensy/zzn/doping-Ga2O3/H_O1/initial_structure/q-1
Succeed job 667.host5: /data2/home/chensy/zzn/doping-Ga2O3/H_O2/initial_structure/q+2

...
```

Calculate the correction for the charged defects:

After finishing the calculations of all the charged defects (except for interstitial defects with high energy), DEC module will calculate the FNV correction (according to the parameter *correction = FNV* in `dasp.in`), and then the formation energies and transition energy levels are also calculated. The specific data of the corrections and formation energies of different charge states of each defect are recorded in file `3dec.out` :

```
...
The formation energy (neutral) of H_O1 at p1 is 1.84 eV
The formation energy (neutral) of H_O1 at p2 is 7.42 eV
The FNV correction (q = 2) E_correct =  -0.247 eV
The transition level (0/2+) above VBM: 2.627 eV
The FNV correction (q = 1) E_correct =  0.082 eV
The transition level (0/+) above VBM: 4.818 eV
The FNV correction (q = 3) E_correct =  -0.075 eV
The transition level (0/3+) above VBM: 1.739 eV
The FNV correction (q = -1) E_correct =  -0.056 eV
The transition level  (-/0) above VBM: 4.769 eV

...

The static calculation of /data2/home/chensy/zzn/doping-Ga2O3/dec/Doping_H/H_i/random3/
↪initial_structure/q0/static is skipped, skip formation energy calculation

...
```

All the data related to formation energies and transition energy levels are also recorded in the file `defect.log` under each defect's corresponding directories.

Output the image of formation energy:

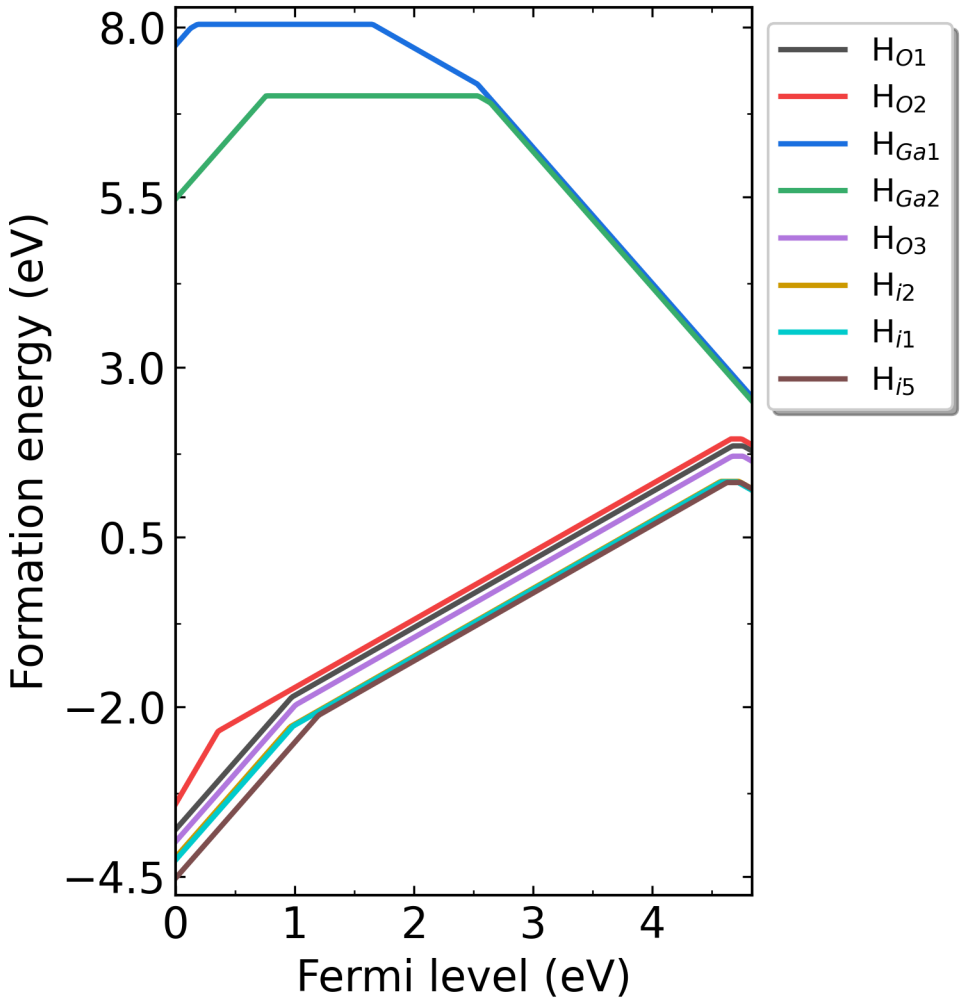
Finally, DEC will automatically output the image of defect formation energy v.s. Fermi level by using all the corrected defect formation energies of Ga2O3 at two chemical potentials. As shown in below:

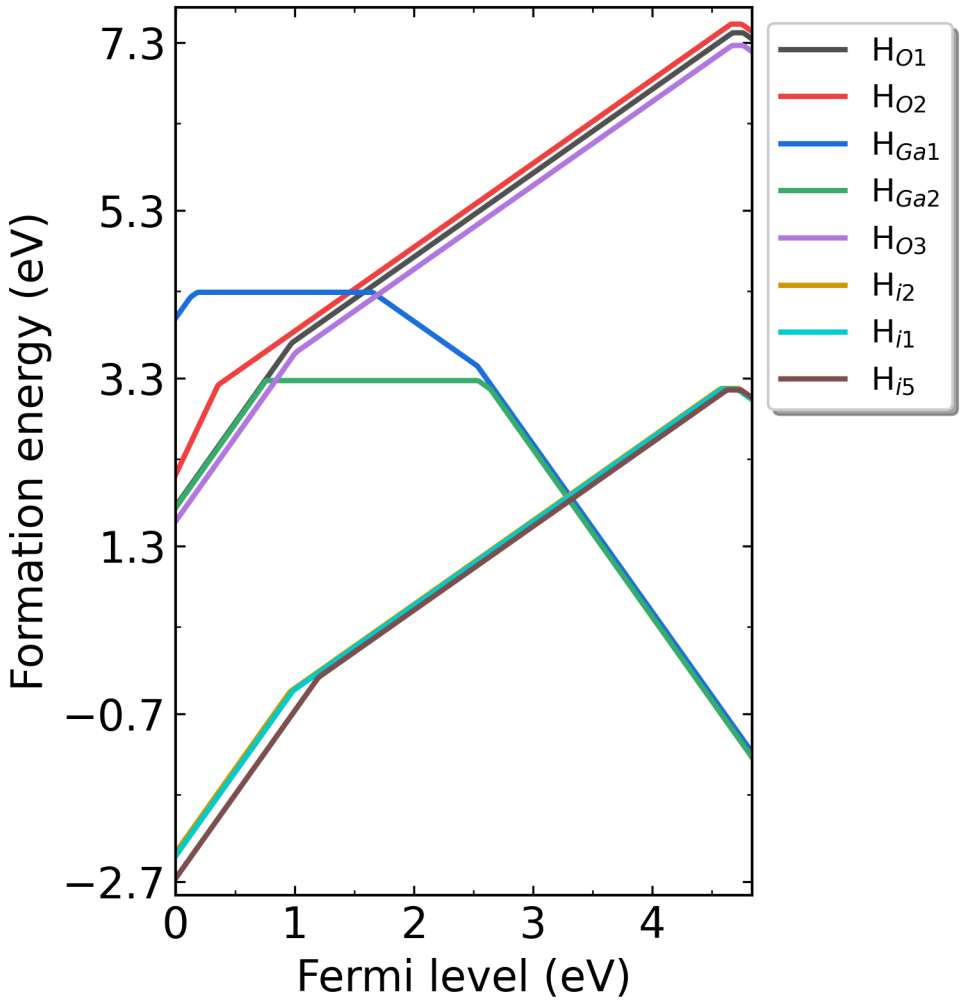
Fig: Formation energies of H-related defects in H-doped Ga2O3 as functions of Fermi level at p1 point (Ga-rich condition).

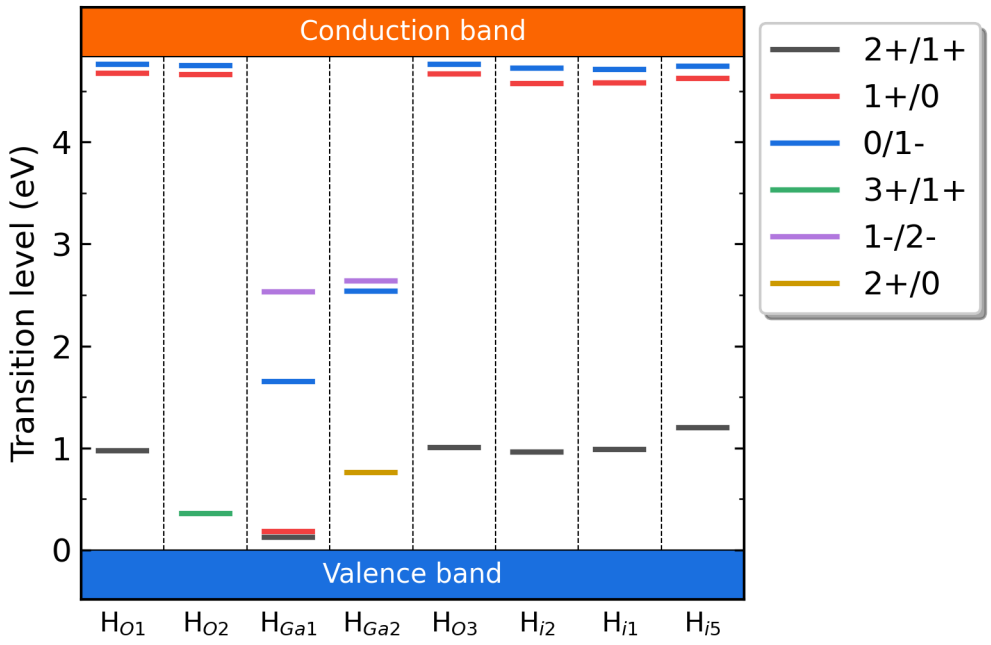
Fig: Formation energies of H-related defects in H-doped Ga2O3 as functions of Fermi level at p2 point (O-rich condition).

Meanwhile, the image of transition energy levels for each defect is also output, as shown in below:

Fig: The charge-state transition energy levels of H-related defects in H-doped Ga2O3.


8.0
5.5
3.0
0.5
−2.0
−4.5
Formation energy (eV)
0
1
2
3
4
Fermi level (eV)
H_{O1}
H_{O2}
H_{Ga1}
H_{Ga2}
H_{O3}
H_{i2}
H_{i1}
H_{i5}


7.3
5.3
3.3
1.3
−0.7
−2.7
Formation energy (eV)
0
1
2
3
4
Fermi level (eV)
H_{O1}
H_{O2}
H_{Ga1}
H_{Ga2}
H_{O3}
H_{i2}
H_{i1}
H_{i5}



5.3.4 5.3.4 DDC -- defect density and Fermi level calculations

5.3.4.1 Run DDC module

After finishing the DEC module, return to the directory doping-Ga2O3 and use the command *dasp 4* to execute the DDC module. No additional operation is required while waiting for the program to complete.

5.3.4.2 Workflow of DDC module

Summarize the defect-related data:

Firstly, DDC module will judge which defects have been finished calculating based on the output of DEC module, and take all these defects into consideration in the DDC calculation. Then, DDC will search for the output information about each defect as formation energy, transition energy levels, and degeneracy factor. Finally, summarize all the data and write in the file `DefectParams.txt` .

`4ddc.out` is the log file of DDC module:

```
############ Collecting information from DEC ############
Read defect types from DEC calculation successfully.
Defects considered in DDC calculation: ['H_O1', 'H_O2', 'H_Ga1', 'H_Ga2', 'H_O3', 'H_i-2
↪', 'H_i-1', 'H_i-5']
Chemical potentials change from p1 to p2.
```

(continues on next page)

(continued from previous page)

```
Calculate gq for defect in each charge state.
Calculate Nsites for H_O1: 5.727808e+22 cm^-3.
Calculate Nsites for H_O2: 5.727808e+22 cm^-3.
Calculate Nsites for H_Ga1: 3.818539e+22 cm^-3.
Calculate Nsites for H_Ga2: 3.818539e+22 cm^-3.
Calculate Nsites for H_O3: 5.727808e+22 cm^-3.
Calculate Nsites for H_i-2: 3.182116e+22 cm^-3.
Calculate Nsites for H_i-1: 3.182116e+22 cm^-3.
Calculate Nsites for H_i-5: 3.182116e+22 cm^-3.
############ Collecting information from DEC ############
```

Below is the file `DefectParams.txt` :

```
1000 300
0.230000 4.210000
4.836382
H_O1 5.727808e+22 2 4.818 1 2.627 2 1.739 1 x x 4.769 1 x x x x x x 1.840000 7.420000
H_O2 5.727808e+22 2 4.812 1 2.461 2 1.811 1 1.36 2 4.768 1 x x x x x x 1.942000 7.522000
H_Ga1 3.818539e+22 1 0.466 2 0.373 1 0.332 2 0.336 1 1.41 2 1.647 1 2.779 2 3.312 1 8.
↪045000 4.325000
H_Ga2 3.818539e+22 1 0.857 2 0.756 1 0.571 2 0.374 1 2.307 2 2.163 1 3.112 2 3.554 1 6.
↪993000 3.272000
H_O3 5.727808e+22 2 4.815 1 2.622 2 1.736 1 1.287 2 4.771 1 x x x x x x 1.688000 7.268000
H_i-2 3.182116e+22 2 4.738 1 2.571 2 1.708 1 1.266 2 4.733 1 x x x x x x 1.318000 3.
↪178000
H_i-1 3.182116e+22 2 4.747 1 2.577 2 1.712 1 1.233 2 4.719 1 x x x x x x 1.311000 3.
↪171000
H_i-5 3.182116e+22 2 4.779 1 2.581 2 1.592 1 1.23 2 4.741 1 x x x x x x 1.301000 3.160000
```

Self-consistent calculation under growth temperature:

The DDC module will calculate the dopant and carrier densities at the temperature T=1000 K, and obtain the Fermi level by self-consistently under the charge neutralization condition.

Self-consistent calculation under working (measuring) temperature:

The DDC module will recalculate the defect/dopant and carrier densities at the temperature T=300 K, and re-obtain the Fermi level by self-consistently under the charge neutralization condition.

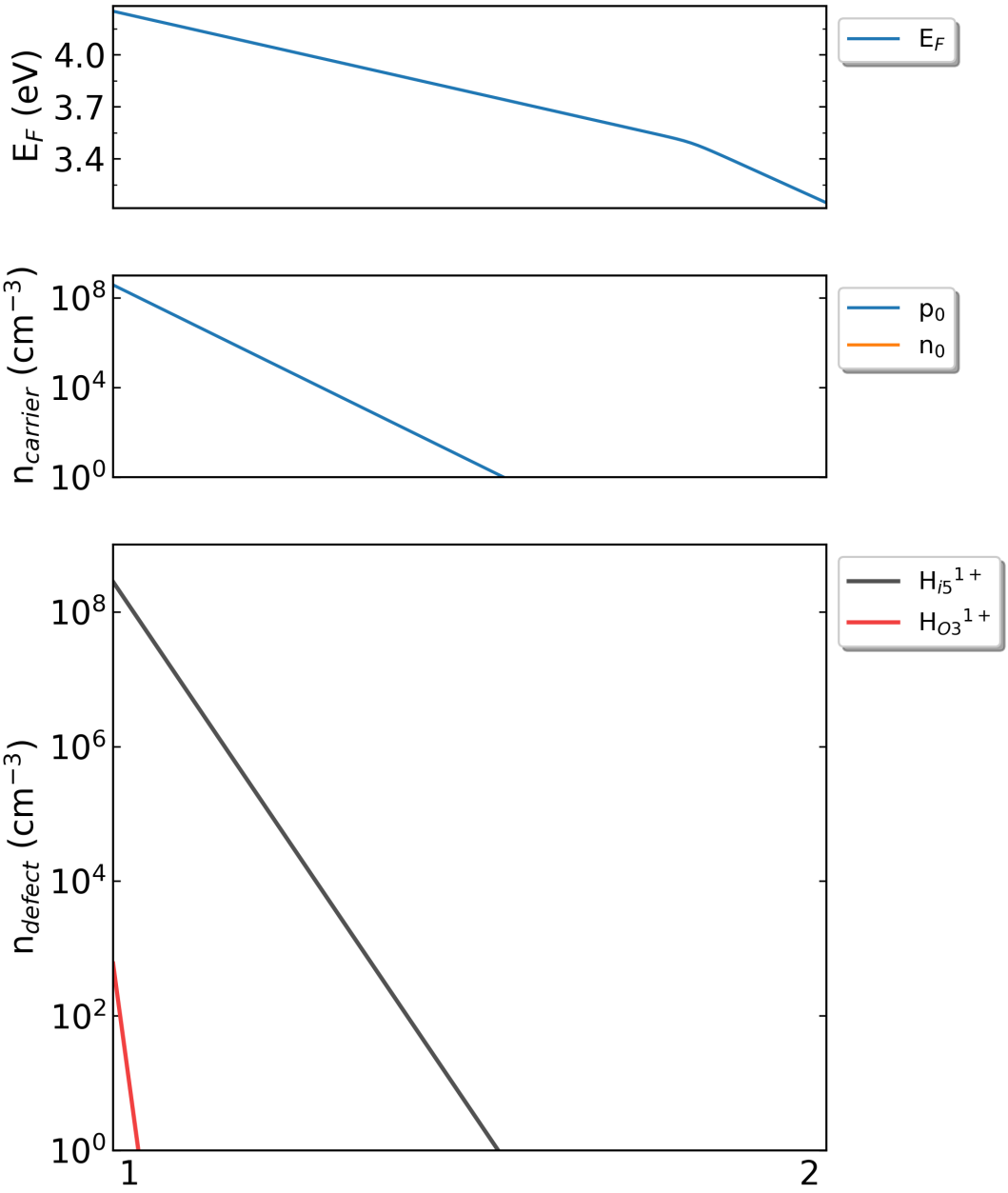
Output defect density:

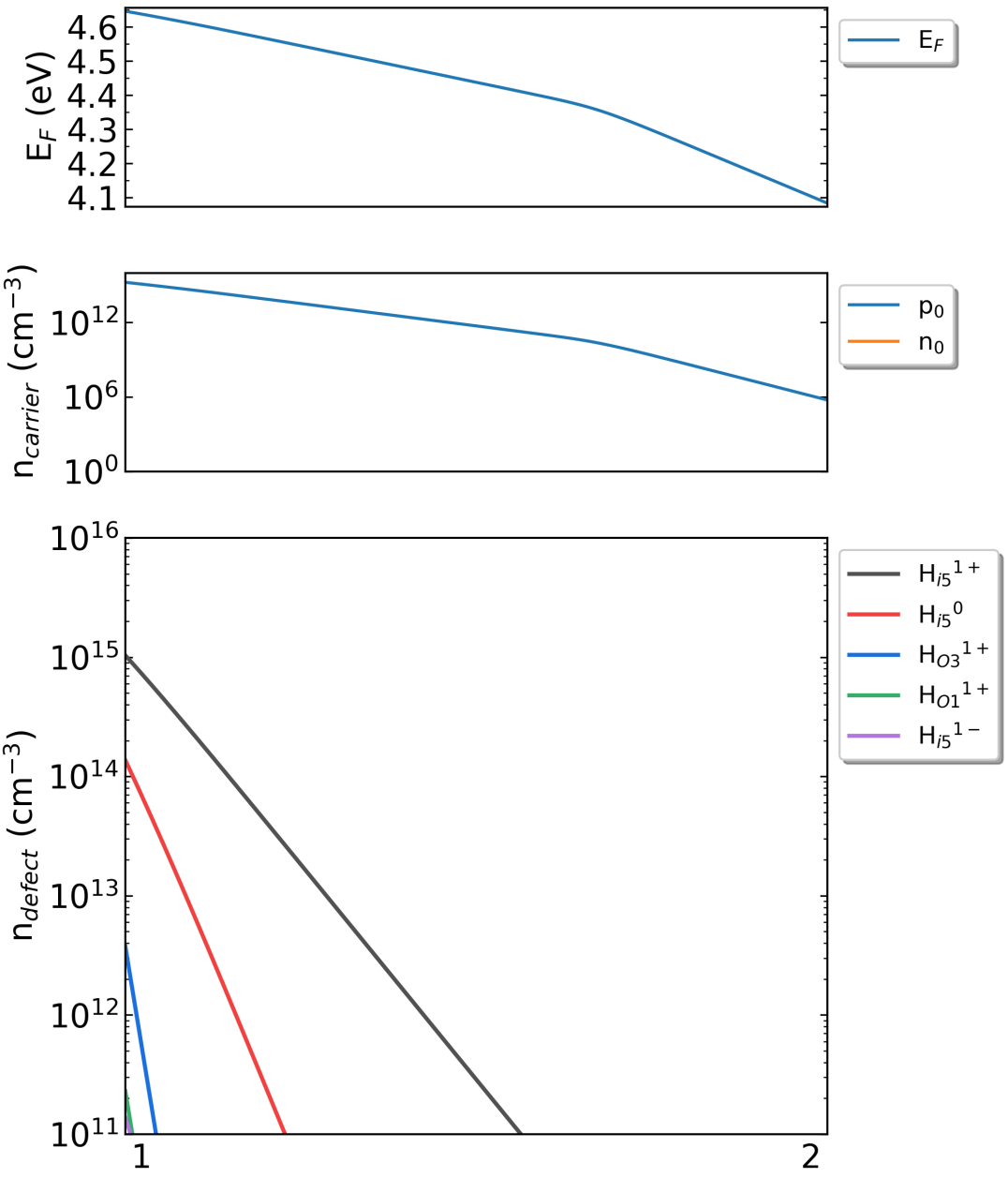
DDC module will output three files in the directory doping-Ga2O3/ddc: `Fermi.dat` , `Carrier.dat` , `Defect_charge.dat` , which can be plotted using Origin. In addition, DDC also can automatically generate image file `density.png` based on these three files, as follows:

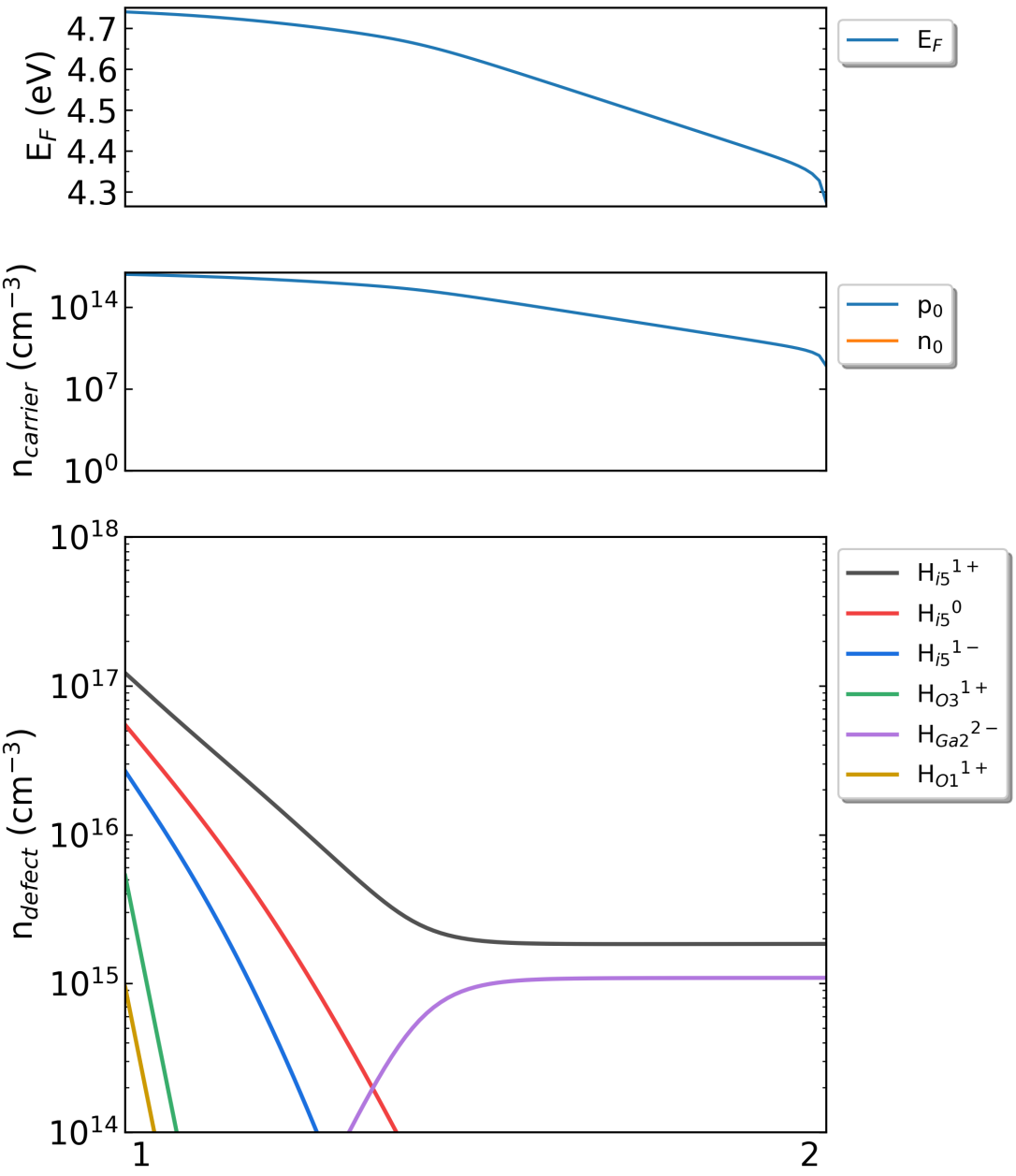
Fig: The Fermi level, electron and hole carrier densities, and defect densities in H-doped Ga2O3 as functions of the chemical potentials (from Ga-rich to O-rich) with a growth temperature is 300 K.

Fig: The Fermi level, electron and hole carrier densities, and defect densities in H-doped Ga2O3 as functions of the chemical potentials (from Ga-rich to O-rich) with a growth temperature is 650 K.

Fig: The Fermi level, electron and hole carrier densities, and defect densities in H-doped Ga2O3 as functions of the chemical potentials (from Ga-rich to O-rich) with a growth temperature is 1000 K.


E_F (eV)
4.0
3.7
3.4
E_F
n_carrier (cm^-3)
10^8
10^4
10^0
p_0
n_0
n_defect (cm^-3)
10^8
10^6
10^4
10^2
10^0
1
2
H_i5^1+
H_O3^1+


4.6
4.5
4.4
4.3
4.2
4.1
E_F (eV)
E_F
10^{12}
10^{6}
10^{0}
$n_{carrier}$ (cm^{-3})
p_0
n_0
10^{16}
10^{15}
10^{14}
10^{13}
10^{12}
10^{11}
n_{defect} (cm^{-3})
1
2
H_{i5}^{1+}
H_{i5}^{0}
H_{O3}^{1+}
H_{O1}^{1+}
H_{i5}^{1-}


4.7
4.6
4.5
4.4
4.3
E_F (eV)
E_F
10^{14}
10^{7}
10^{0}
$n_{carrier}$ (cm^{-3})
p_0
n_0
10^{18}
10^{17}
10^{16}
10^{15}
10^{14}
n_{defect} (cm^{-3})
1
2
H_{i5}^{1+}
H_{i5}^{0}
H_{i5}^{1-}
H_{O3}^{1+}
H_{Ga2}^{2-}
H_{O1}^{1+}

5.4 5.4 The calculation of intrinsic defect in ZnGeP2

ZnGeP2 is a nonlinear optical material, and its application is limited by the existence of the optical absorption peaks in the band gap, which are believed to be related to point defects. Therefore, it is necessary to carry out a theoretical calculation of the point defect properties in ZnGeP2 and analyze the origin of its absorption peaks under different preparation environments.
The following is the example of using DASP to calculate intrinsic defects in ZnGeP2:

5.4.1 5.4.1 PREPARE --- Prepares for calculation

5.4.1.1 POSCAR and dasp.in

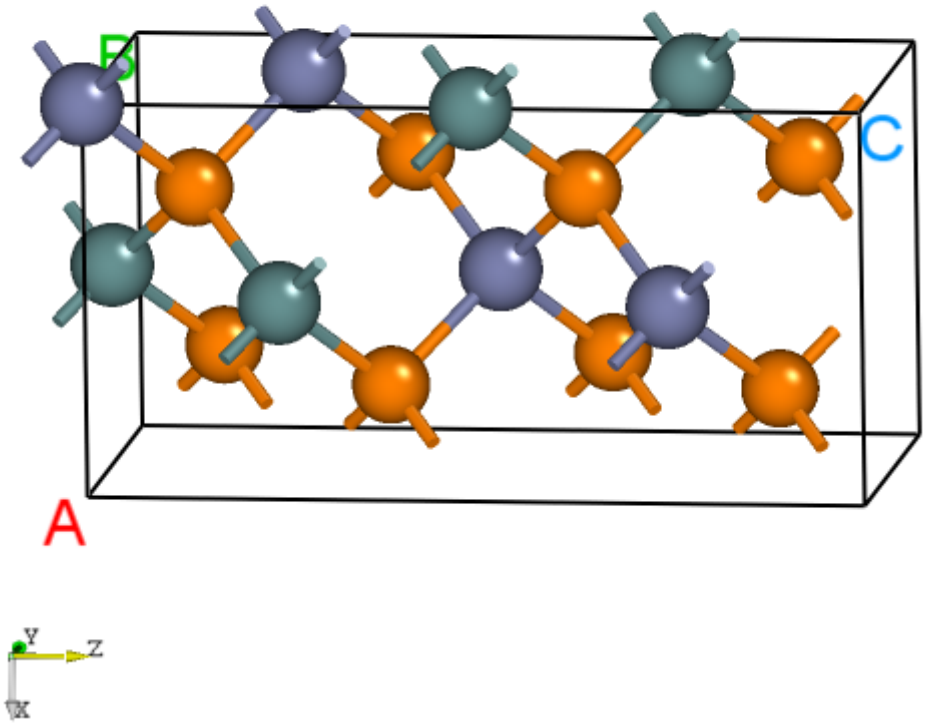
Find the `POSCAR` for ZnGeP2, and use VASP to optimize its lattice parameters or modify the lattice to match the experimental measurements (Users need to do it manually). The information is as follows:

```
Zn4 Ge4 P8
1.0
5.468                0.0000000000         0.0000000000
0.0000000000         5.468                0.0000000000
0.0000000000         0.0000000000         10.745
Zn   Ge    P
4    4    8
Direct
0.000000000          0.500000000          0.250000000
0.000000000          0.000000000          0.000000000
0.500000000          0.000000000          0.750000000
0.500000000          0.500000000          0.500000000
0.500000000          0.000000000          0.250000000
0.500000000          0.500000000          0.000000000
0.000000000          0.500000000          0.750000000
0.000000000          0.000000000          0.500000000
0.250000000          0.754127026          0.375000000
0.745872974          0.750000000          0.125000000
0.254126996          0.250000000          0.125000000
0.750000000          0.245873004          0.375000000
0.750000000          0.254126996          0.875000000
0.245873004          0.250000000          0.625000000
0.754127026          0.750000000          0.625000000
0.250000000          0.745872974          0.875000000
```

Using the crystal visualization software, the structure is shown in below:

The structure of ZnGeP2.

Write the required parameters in `dasp.in` :


B
C
A
Y
Z
X

```
############### Job Scheduling ###############
cluster = SLURM      # (job scheduling system)
node_number = 4          # (number of node)
core_per_node = 32       # (core per node)
queue = normal            # (name of queue/partition)
max_time = 24:00:00         # (maximum time for a single DFT calculation)
vasp_path_dec = /opt/vasp.5.4.4/bin/vasp_gam    #  (path of VASP)
vasp_path_tsc = /opt/vasp.5.4.4/bin/vasp_std
vasp_path_cdc=/opt/vasp-optics/bin/vasp_gam
job_name = submit_job     # (name of script)
potcar_path = /opt/POT/potpaw_PBE     # (path of pseudopotentials)
max_job = 5

############### TSC Module ###############
database_api = ******************** # (str-list type)

############### DEC Module ###############
level = 2    # (level=1: PBE+PBE; level=2: PBE+HSE; level=3: HSE+HSE)
min_atom = 180
max_atom = 200
intrinsic = T    # (default: T)
correction = FNV    # (default: none)
epsilon = 12.3
Eg_real = 2.06    # (experimental band gap)

############### DDC Module ###############
ddc_temperature = 1300 300
ddc_mass = 0.36 0.54
ddc_path = 1 2

############### CDC Module ################
cdc_defect = Ge_Zn1
cdc_job = pl / radiative_rate
cdc_charge = 0 1
cdc_band = 864 865
cdc_temperature = 300
spin_channel = 2
refractive_index = 2.38
```

Next, all the parameters listed in `dasp.in` will be described:

```
cluster = SLURM
# The system of the used cluster is SLURM
```

```
node_number = 4
# 4 nodes are used for each calculation.
```

```
core_per_node = 32
# 32 cores are used for each node, so 4*32=128 cores are used in total for each␣
↪calculation.
```

```
queue = normal
# The queue named "normal" is used to carry out calculations. Therefore, users need to␣
↪make sure the queue name, nodes, and cores of clusters before configuring dasp.in.
```

```
max_time = 24:00:00        # (maximum time for a single DFT calculation)
# The queue named "normal" is used to carry out calculations. Therefore, users need to␣
↪make sure the queue name, nodes, and cores of clusters before configuring dasp.in.
```

```
vasp_path_dec = /opt/vasp.5.4.4/bin/vasp_gam   #  (path of VASP)
vasp_path_tsc = /opt/vasp.5.4.4/bin/vasp_std
vasp_path_cdc = /opt/vasp-optics/bin/vasp_gam
# The VASP_std version is used for TSC calculations, and the VASP_gam version is used␣
↪for DEC calculations. the VASP_gam version is specified in CDC calculation to␣
↪calculate the carrier transition matrix element.
```

```
job_name = submit_job    # (name of script)
# The submission script, named " submit_job" and can be set arbitrarily.
```

```
potcar_path = /opt/POT/potpaw_PBE    # (path of pseudopotentials)
# path of pseudopotentials
```

```
max_job = 5
# the allowed maximum number of jobs at the same time
```

```
database_api = ******************** # (str-list type)
# using to visit the Materials Project database
```

```
level = 2   # (level=1: PBE+PBE; level=2: PBE+HSE; level=3: HSE+HSE)
# using to visit the Materials Project database
```

```
min_atom = 180
max_atom = 200
# The number of atoms within the generated supercell that we want is between 180 and 200,
↪ and as far as possible to make a=b=c and a b c.
```

```
intrinsic = T   # (default: T)
# Generate intrinsic defects, V_Zn, V_Ge,V_P, Zn_Ge, Zn_P, Ge_Zn, Ge_P, P_Zn, P_Ge, Zn_i,
↪ Ge_i, and P_i.i
```

```
correction = FNV   # (default: none)
# The corrections for charged defect adopt FNV correction.
```

```
epsilon = 12.3
# The dielectric constant of ZnGeP2 is 12.3.
```

```
Eg_real = 2.06   # (experimental band gap)
# The experimental band gap of ZnGeP2 is about 2.06 eV, DASP will adjust AEXX in INCAR␣
↪to make the band gap of the supercell without defect equal to 2.06 eV.
```

```
ddc_temperature = 1300 300
# the growth temperature set to 1300 K and the working temperature set to 300 K.
```

```
ddc_mass = 0.36 0.54
# electron effective mass set to 0.36 and hole effective mass set to 0.54.
```

```
ddc_path = 1 2
# set
```

```
cdc_defect = Ge_Zn1
# calculate the relevant properties of Ge_Zn1
```

```
cdc_job = pl / radiative_rate
# calculate the PL spectrum or radiative capture coefficient
```

```
cdc_charge = 0 1
# The defect state changes from the neutral state to the ionized state with +q, which is␣
↪a hole transition.
```

```
cdc_band = 864 865
# hole transition, a hole transfer to the defect state from the valence band maximum, i.
↪e. from the 864th band transfer to the 865th band.
```

```
cdc_temperature = 300
# calculate the defect properties at 300K in CDC module
```

```
spin_channel = 2
# The spin of the carrier is spin down.
```

```
refractive_index = 2.38
# The refractive index of ZnGeP2 is 2.38
```

5.4.1.2 Use DASP to generate the required input files

Create a new directory ZnGeP2, then prepare the files, `POSCAR` and `dasp.in` , mentioned above in the directory ./ZnGeP2/. Next, execute `dasp 1` to start PREPARE module and no additional operation is needed thereafter. DASP will output file `1prepare.out` to record the running log of the module.

5.4.1.3 Workflow of PREPARE module

Generate supercell:

Firstly, the program will automatically find the optimal supercell (as far as possible to make a=b=c and a b c) based on the parameters *min_atom=180* and *max_atom=200* and output `POSCAR` for the supercell. The following are the structure messages of supercell `POSCAR_nearlycube` expanded by ZnGeP2 primitive cell.

```
Cubic_cell
1.0
16.4040000000 0.0000000000 0.0000000000
0.0000000000 15.3313770093 0.0000000000
0.0000000000 0.2701043094 15.3289975100
Zn Ge P
48 48 96
Direct
0.0000000000 1.0000000000 0.2500000000
0.0000000000 0.0000000000 0.0000000000
0.1666666666 0.6250000000 0.3750000000
0.1666666666 0.8750000000 0.3750000000
...
```

The supercell generated by DASP is shown in below:

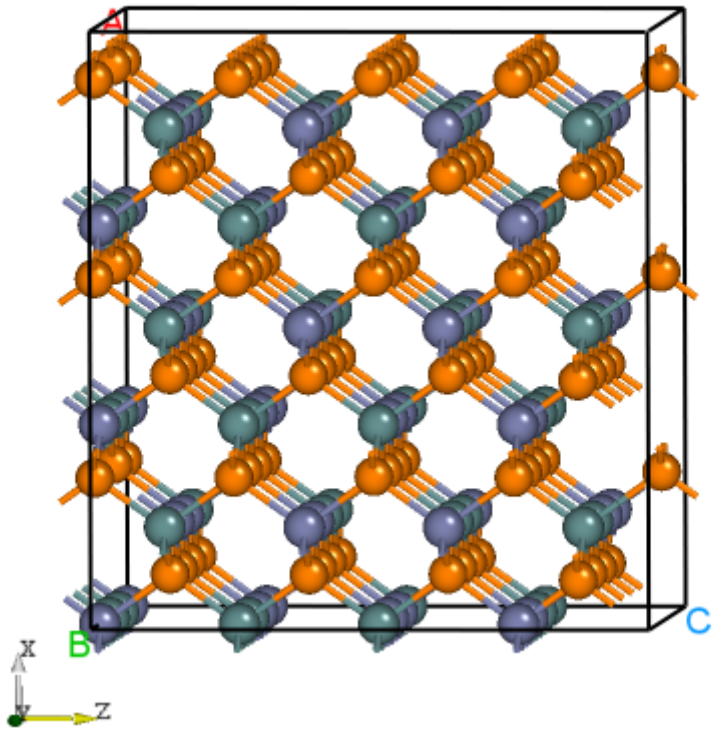


Fig: The structure of ZnGeP2 supercell.

Madelung constant calculation:

Secondly, according to the generated supercell, the program will execute Madelung constant calculation

which describes the Coulomb interaction between charged defect and periodic image charge. (use for Lany-Zunger correction)

After finishing the above two steps calculation, the output of `1prepare.out` is as follows (**** indicates the job ID of the calculation):

```
############ Prepare Files module start ############

Read the structure file POSCAR you provided
Get the refined cell POSCAR_refined from POSCAR
Generate the nearlycube cell POSCAR_nearlycube from POSCAR
Generate job script through dasp.in parameters
Generate single-point KPOINTS
Generate pseudopotential file POTCAR through potcar_dir you set
Generate commonly used vasp input file INCAR
Start the madelung constant calculation
Generate the madelung calculation directory
Generate madelung calculation POSCAR
Generate madelung calculation POTCAR
Generate madelung calculation INCAR
Generate madelung calculation KPOINTS
Generate madelung calculation job script
Job ******** submitted: /home/test/ZnGeP2/dec/madelung/static
Succeed job ********: /home/test/ZnGeP2/dec/madelung/static
The madelung constant calculation completed
The madelung constant = 2.833
```

HSE exchange proportion calculation:

According to the generated supercell, the program will perform HSE static calculations with *AEXX=0.25* and *AEXX=0.3* respectively to determine the value of *AEXX* which can make the obtained band gap match *Eg_real = 2.06* based on the slope. Therefore, after those calculations are completed, the contents in the directory ZnGeP2/dec/AEXX/ are as follows:

```
cd ./dec/AEXX
ls
0.25  0.26795555051593156  0.3  AEXX.list
```

It indicates that the band gap of the CdTe supercell is 2.06 eV when *AEXX=0.27* (two decimal places), and write this parameter into `INCAR` . Meanwhile, the log can be seen from `1prepare.out` as follows (**** indicates the job ID of the calculation):

```
Start the HSE parameter AEXX calculation
Job ******** submitted: /home/test/ZnGeP2/dec/AEXX/0.25/static
Job ******** submitted: /home/test/ZnGeP2/dec/AEXX/0.3/static
Succeed job ********: /home/test/ZnGeP2/dec/AEXX/0.25/static
```

(continues on next page)

(continued from previous page)

```
Succeed job ********: /home/test/ZnGeP2/dec/AEXX/0.3/static
Job ******** submitted: /home/test/ZnGeP2/dec/AEXX/0.26795555051593156/static
Succeed job ********: /home/test/ZnGeP2/dec/AEXX/0.26795555051593156/static
The HSE parameter AEXX calculation completed
The HSE parameter AEXX = 0.27
level = 2: Generate PBE relax vasp input file INCAR-relax
level = 2: Generate HSE static vasp input file INCAR-static
```

Optimize the ionic position of the host supercell:

The last step in PREPARE module is to optimize the ionic position of the host supercell according to *level=2* (PBE relax). The optimized file is `POSCAR_final` in the directory ZnGeP2/dec/relax. At the same time, the sign of the end of DASP operation can be seen in `1prepare.out` , and it also tells us that we need to do the TSC module calculation in the next step (**** indicates the job ID of the calculation).

```
Start the POSCAR_nearlycube relax calculation
Generate the POSCAR_nearlycube relax directory
Job ******** submitted: /home/test/ZnGeP2/dec/relax
Succeed job ********: /home/test/ZnGeP2/dec/relax
The POSCAR_nearlycube relax calculation completed
Get the final structure POSCAR_final

############ Prepare Files module end ############

DASP-PREPARE finished, please run DASP-TSC next
```

5.4.2 5.4.2 TSC -- thermodynamic stability and chemical potential calculations

5.4.2.1 Run TSC module

The directory ZnGeP2/dec will be created when using the command `dasp 1` to execute PREPARE module, and generate file `1prepare.out` in this directory. After finishing the program, there has the corresponding completion flag in `1prepare.out` . Then, enter the directory ZnGeP2/dec and confirm that the parameters in `INCAR-relax` and `INCAR-static` are feasible. (Users can modify INCAR, and DASP will make subsequent calculations based on the INCAR in this directory.)

Once confirm that the PREPARE module is finished, return to the directory ZnGeP2 and use the command `dasp 2` to execute the TSC module. Similarly, the TSC module will create a directory named tsc under the directory ZnGeP2, in which contains the output of the TSC program, including every calculation directory and the running log file `2tsc.out` . No additional operation is required while waiting for the program to complete.

5.4.2.2 Workflow of TSC module

The total energy calculation of the host structure (the parameters are consistent with MP database):

TSC module will use the same input parameters (INCAR, KPOINTS, POTCAR) with the `Materials Project` database to perform structural relaxation and static calculation on the primitive cells given by the user. Therefore, the calculated total energy is comparable to that of the MP database. This step is to obtain the **key hetero-phases** that limit the stability of ZnGeP2. In the directory, we can see:

```
cd tsc
cd ZnGeP2/
ls
relaxation1  relaxation2  static
```

The running log also can be seen from the ZnGeP2/tsc/2tsc.out, that is, the steps such as generating input files, relaxation1, relaxation2, static and data extraction.

The judgement of key hetero-phases compounds:

The TSC module will search for all the secondary compounds that compete with ZnGeP2 in the MP database. And compare the total energy of ZnGeP2 calculated in the previous step with that of the hetero-phases extracted from the database to confirm ZnGeP2 is **thermodynamically stable**.

Subsequently, the program will automatically download the key hetero-phases compounds that can limit the thermodynamic stability of ZnGeP2. Ge, P, Zn3P2, ZnP2, and Zn are considered in this case. The relevant information can be seen in `2tsc.out` :

```
...
analysing the thermodynamic stability of ZnGeP2.
key phases of ZnGeP2 are: Ge P Zn3P2 ZnP2 Zn .
file key_phases_info_recalc.yaml generated.
analysing of ZnGeP2 is done.
...
```

The total energy calculation of the host and hetero-phase compounds:

After the key hetero-phase compounds are confirmed, TSC will calculate the total energy of ZnGeP2, Ge, P, Zn3P2, ZnP2, and Zn by using the parameter (AEXX) obtained from PREPARE module. `2tsc.out` is as follows:

```
...
Job ******** submitted: /home/test/ZnGeP2/tsc/ZnGeP2/static_recalc
Job ******** submitted: /home/test/ZnGeP2/tsc/Ge/static_recalc
Job ******** submitted: /home/test/ZnGeP2/tsc/P/static_recalc
Job ******** submitted: /home/test/ZnGeP2/tsc/Zn3P2/static_recalc
Job ******** submitted: /home/test/ZnGeP2/tsc/ZnP2/static_recalc
Job ******** submitted: /home/test/ZnGeP2/tsc/Zn/static_recalc
Succeed job  ********: /home/test/ZnGeP2/tsc/ZnGeP2/static_recalc
Succeed job  ********: /home/test/ZnGeP2/tsc/Ge/static_recalc
Succeed job  ********: /home/test/ZnGeP2/tsc/P/static_recalc
Succeed job  ********: /home/test/ZnGeP2/tsc/Zn3P2/static_recalc
Succeed job  ********: /home/test/ZnGeP2/tsc/Zn/static_recalc
Succeed job  ********: /home/test/ZnGeP2/tsc/ZnP2/static_recalc
...
```

The chemical potential calculation:

Calculating the formation energy and stable chemical potential region of ZnGeP2 based on the calculated total energy. As ZnGeP2 is a ternary compound, TSC module will give the endpoint of four chemical potentials, and write them into `dasp.in` :

```
# The orders are consistent with the order of elements in POSCAR, i.e. the first column␣
↪is Zn, the second column is Ge, and the third column is P.
E_pure = -2.0283 -5.9739 -7.3365
p1 = -0.1456 0.0 -0.4672
p2 = -1.08 0.0 0.0
p3 = -0.9207 -0.1593 0.0
p4 = -0.2252 -0.1593 -0.3478
```

The output after the program is completed can be seen in `2tsc.out` :

```
dir '2d-figures','3d-figures','ori_data_MP' ready.  try to read file: 'calc_list.yaml'.
analysing the thermodynamic stability of ZnGeP2.
key phases of ZnGeP2 are: Ge P Zn3P2 ZnP2 Zn .
analysing of ZnGeP2 is done.

DASP-TSC finished
```

For the ternary and multinary compounds, TSC module will output the image of the stable region and the chemical potential at the endpoint of the stable region. In this directory, it can be seen:

```
cd tsc
cd 2d-figures/
ls
fig-ZnGeP2.png  fig-ZnGeP2_recalc.png  stable_2d.out stable_recalc_2d.out
```

The four files under the directory ZnGeP2/tsc/2d-figures/ are the images of the stable region plotted during the two calculations and analyses, and the chemical potential at each endpoint in the images.

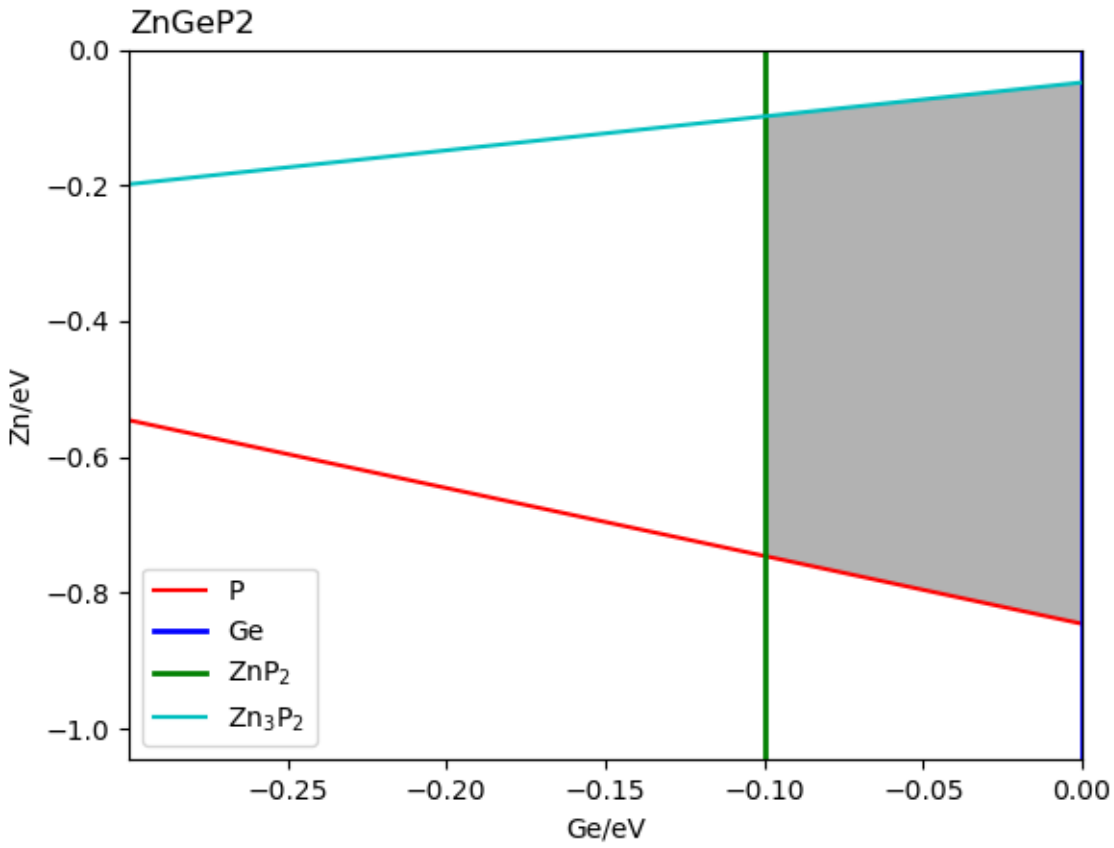
Check out the files `stable_2d.out` and `fig-ZnGeP2.png` . The horizontal and vertical coordinates of `fig-ZnGeP2.png` are the chemical potential of the elements marked in the figure, and the shaded area is the stable region of the host compound. Each line on the boundary is the chemical potential curve at the critical condition where the marked material can or can not form. This is the image output from the first calculation and analysis process.

```
     Ge        Zn         P
-------  -------  -------
-0.0999  -0.7457    0
0        -0.8457    0
0        -0.0484  -0.3986
-0.0999  -0.0983  -0.3237
```

Fig: The stable chemical potential region of ZnGeP2. (comes from the MP database)

Check out the files `stable_recalc_2d.out` and `fig-ZnGeP2_recalc.png` . They are the image and data output from the second calculation and analysis process.

```
     Ge        Zn         P
-------  -------  -------
-0.1593  -0.9207    0
0        -1.08      0
0        -0.1456  -0.4672
-0.1593  -0.2252  -0.3478
```


ZnGeP2
0.0
−0.2
−0.4
−0.6
−0.8
−1.0
Zn/eV
−0.25
−0.20
−0.15
−0.10
−0.05
0.00
Ge/eV
P
Ge
ZnP2
Zn3P2

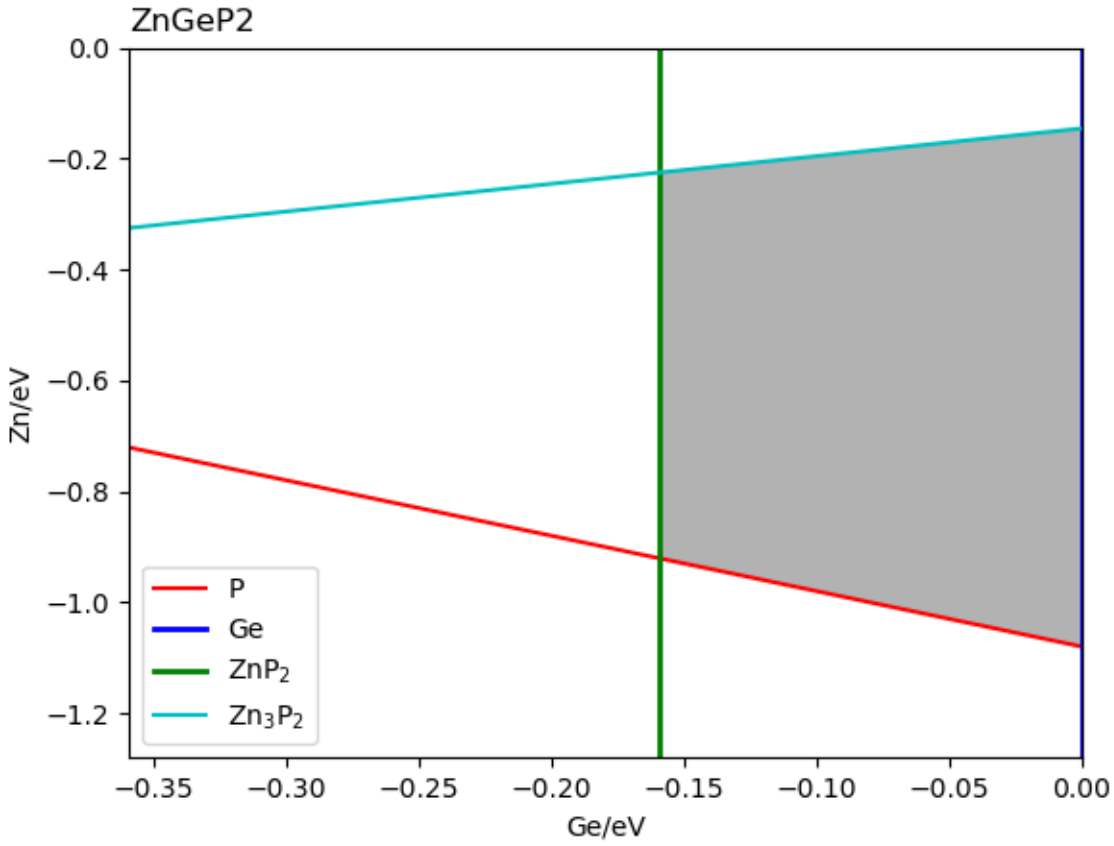

ZnGeP2
0.0
−0.2
−0.4
−0.6
−0.8
−1.0
−1.2
Zn/eV
−0.35
−0.30
−0.25
−0.20
−0.15
−0.10
−0.05
0.00
Ge/eV
P
Ge
ZnP2
Zn3P2

Fig: The stable chemical potential region of ZnGeP2. (comes from the DFT calculation)

5.4.3 5.4.3 DEC -- the calculations of defect formation energy and transition energy level

5.4.3.1 Run DEC module

The directory ZnGeP2/tsc will be created when using the command `dasp 2` to execute TSC module, and generate file `2tsc.out` in this directory. After finishing the TSC module, there has the corresponding completion flag in `2tsc.out` . Then, open the file `dasp.in` under the directory ZnGeP2/dasp.in to confirm the chemical potential already has been written.

Once confirm that the TSC module is finished, return to the directory ZnGeP2 and use the command `dasp 3` to execute the DEC module. DEC will output relevant files in the generated directory dec in the first step, including the defect structures, directories, and the log file `3dec.out` . No additional operation is required while waiting for the program to complete.

5.4.3.2 Workflow of DEC module

Generate defects structure:

In order to calculate the defect properties of ZnGeP2, the program will generate the directories and structures corresponding to the neutral defects that need to be calculated based on the user setting in `dasp.in` :

```
cd dec
cd Intrinsic_Defect
ls
Ge_i   Ge_Zn1  Intrinsic_Defect.list  P_i    V_Ge1  V_Zn1   Zn_i
Ge_P1  host    P_Ge1                 P_Zn1  V_P1   Zn_Ge1  Zn_P1
```

At the same time, the output of DEC module can be seen in `3dec.out` as follows:

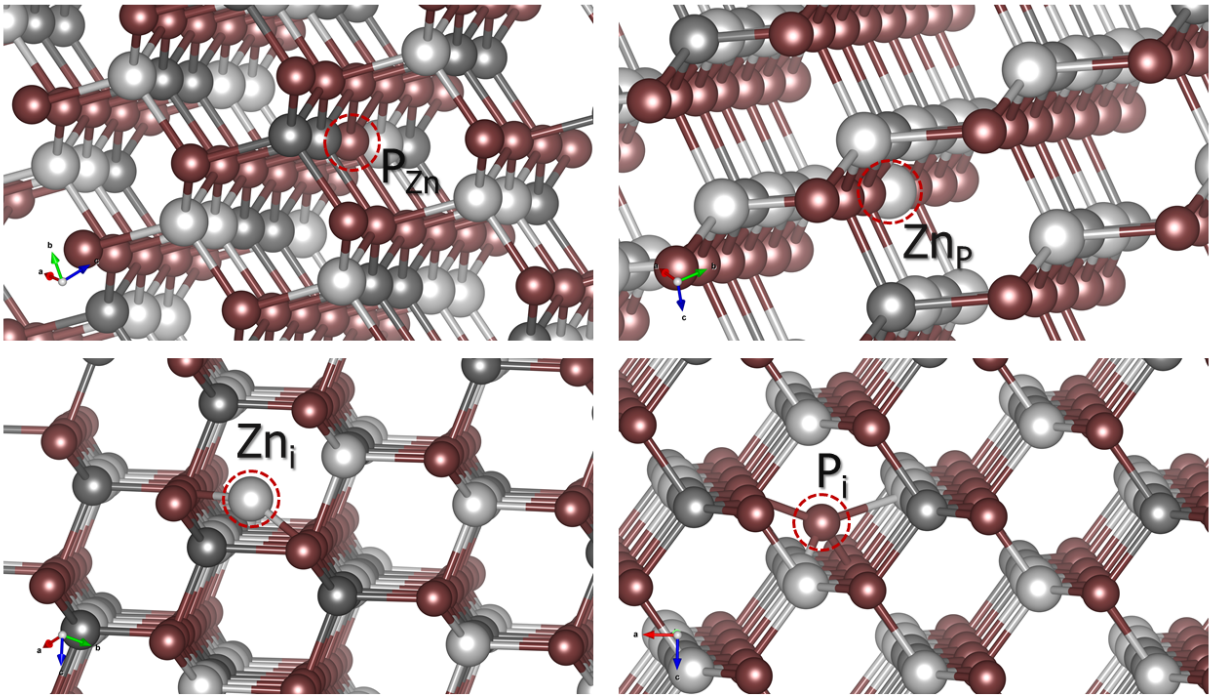
```
############ Neutral Defect module start ############
Make intrinsic defect directory Intrinsic_Defect
Generate host directory in Intrinsic_Defect
Start generating neutral vacancy defect
Generate neutral defect at: V_Zn1/initial_structure/q0
Generate neutral defect at: V_Ge1/initial_structure/q0
Generate neutral defect at: V_P1/initial_structure/q0
Neutral vacancy defect generation completed
Start generating neutral intrinsic antisite defect
Generate neutral defect at: Ge_Zn1/initial_structure/q0
Generate neutral defect at: P_Zn1/initial_structure/q0
Generate neutral defect at: Zn_Ge1/initial_structure/q0
Generate neutral defect at: P_Ge1/initial_structure/q0
Generate neutral defect at: Zn_P1/initial_structure/q0
Generate neutral defect at: Ge_P1/initial_structure/q0
Neutral intrinsic antisite defect generation completed
Start generating neutral intrinsic interstitial defect
Generate neutral defect at: Zn_i/random1/initial_structure/q0
Generate neutral defect at: Zn_i/random2/initial_structure/q0
...
Generate neutral defect at: Ge_i/random1/initial_structure/q0
```

(continues on next page)

(continued from previous page)

```
Generate neutral defect at: Ge_i/random2/initial_structure/q0
...
Generate neutral defect at: P_i/random1/initial_structure/q0
Generate neutral defect at: P_i/random2/initial_structure/q0
...
Neutral intrinsic interstitial defect generation completed

############ Neutral Defect module end ############
```

Among the generated defect structures, part of antisite and possible interstitial defects is shown in below:



Part of defect structures of ZnGeP2.

Submit jobs for all defects with q=0:

After the structures and directories of neutral defects are created, DEC module will automatically supplement the files required for the calculations and submit jobs in turn, so that the number of jobs to be calculated at the same time does not exceed the value of *max_job* in `dasp.in` . The following logs can be seen in file `3dec.out` :

```
############ AutoRun - Neutral Defect module start ############
Job ******** submitted: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/Intrinsic_Defect/P_
↪Zn1/initial_structure/q0
Job ******** submitted: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/Intrinsic_Defect/Zn_
↪P1/initial_structure/q0
Job ******** submitted: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/Intrinsic_Defect/Ge_
↪P1/initial_structure/q0
Job ******** submitted: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/Intrinsic_Defect/P_
↪Ge1/initial_structure/q0
Job ******** submitted: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/Intrinsic_Defect/P_i/
```

(continues on next page)

(continued from previous page)

```
↪random3/initial_structure/q0
Succeed job ********: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/Intrinsic_Defect/Zn_P1/
↪initial_structure/q0
Job ******** submitted: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/Intrinsic_Defect/P_i/
↪random1/initial_structure/q0
Succeed job ********: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/Intrinsic_Defect/Ge_P1/
↪initial_structure/q0
...

############ AutoRun - Neutral Defect module end ############
```

Generate calculation directories for the charged defects:

After finishing the calculations of all the neutral defects, the program will judge the charge states of each defect and generate the corresponding directories and files for the charged defects based on the results of the neutral defects.

```
############ Ionized Defect module start ############
Start generating ionized defects
Ionized defect path: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/Intrinsic_Defect/P_Zn1/
↪initial_structure/q-4
Ionized defect path: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/Intrinsic_Defect/P_Zn1/
↪initial_structure/q-3
...
Ionized defects generation completed

############ Ionized Defect module end ############
```

Submit jobs for the defects with q 0:

After the structures and directories of the charged defects are created, DEC module will automatically supplement the files required for the calculations and submit jobs in turn, so that the number of jobs to be calculated at the same time does not exceed the value of *max_job* in `dasp.in` . The following logs can be seen in file `3dec.out` :

```
############ AutoRun - Ionized Defect module start ############
Job ******** submitted: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/Intrinsic_Defect/P_
↪Zn1/initial_structure/q-1
Job ******** submitted: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/Intrinsic_Defect/P_
↪Zn1/initial_structure/q-4
Job ******** submitted: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/Intrinsic_Defect/P_
↪Zn1/initial_structure/q-2
Job ******** submitted: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/Intrinsic_Defect/P_
↪Zn1/initial_structure/q+3
Job ******** submitted: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/Intrinsic_Defect/P_
↪Zn1/initial_structure/q-3
Succeed job ********: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/Intrinsic_Defect/P_Zn1/
↪initial_structure/q-1
Job ******** submitted: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/Intrinsic_Defect/P_
↪Zn1/initial_structure/q+1
...

############ AutoRun - Ionized Defect module end ############
```

Calculate the correction for the charged defects:

After finishing the calculations of all the charged defects, the program will calculate the formation energies and transition energy levels for each defect. The formation energies, band alignment, image charge correction (LZ/FNV), and the transition energy levels of each defect under different chemical potentials are recorded in the log file `3dec.out` . There are four formation energies corresponding to the four chemical potentials at p1, p2, p3, and p4 of each element provided in file `dasp.in` . The following logs can be seen in file `3dec.out` :

```
############ Formation Energy module start ############
Start the formation energy calculation

The formation energy (neutral) of P_Zn1 at p1 is 3.993075
The formation energy (neutral) of P_Zn1 at p2 is 2.591475
The formation energy (neutral) of P_Zn1 at p3 is 2.750775
The formation energy (neutral) of P_Zn1 at p4 is 3.794075
The FNV correction (q = -1) E_correct =  0.0859854 eV
The transition level  (-/0) above VBM: 1.3758
The FNV correction (q = -4) E_correct =  1.43843 eV
The transition level  (4-/0) above VBM: 2.0373
...

############ Formation Energy module end ############
```

Output the image of formation energy:

After the calculations of formation energy and transition energy level for each kind of defect are completed, DEC module will automatically output the image of defect formation energy and data under different chemical potential conditions, and save it in the directory /dec/Formation_Energy_Intrinsic_Defect/. As four values of the chemical potential of each element are provided in file `dasp.in` , there are also four images and corresponding data.

```
cd dec
cd Formation_Energy_Intrinsic_Defect/
ls
p1.dat  p1.png  p2.dat  p2.png  p3.dat  p3.png  p4.dat  p4.png
```

Users can plot manually based on the data file with .dat format, or refer to the image file with .png format automatically generated by the program. The following logs can be seen in file `3dec.out` :

```
############ Plot Diagram module start ############
Start plotting the diagrams
Generate formation energy diagrams at p1: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/
↪Formation_Energy_Intrinsic_Defect/p1.dat
Generate formation energy diagrams at p2: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/
↪Formation_Energy_Intrinsic_Defect/p2.dat
Generate formation energy diagrams at p3: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/
↪Formation_Energy_Intrinsic_Defect/p3.dat
Generate formation energy diagrams at p4: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/
↪Formation_Energy_Intrinsic_Defect/p4.dat
Generate transition level diagram: /data/home/test/Zn_Ge_P/DASP-test-0128/dec/Transition_
↪Level_Intrinsic_Defect/tl.dat
All diagrams completed

############ Plot Diagram module end ############
```

The four images automatically plotted by DEC module are shown in below:

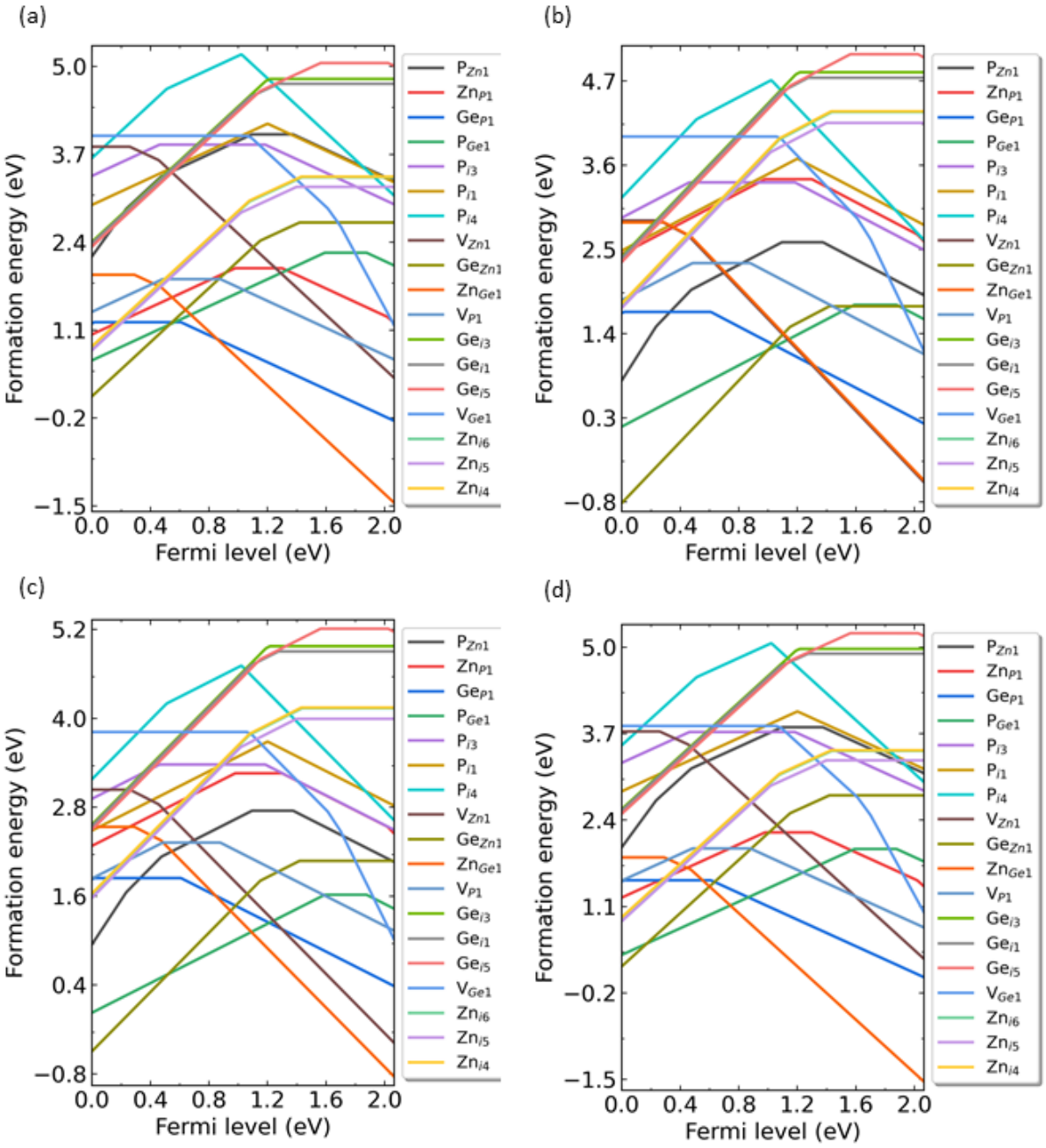


Fig: Formation energies of intrinsic defects in ZnGeP2 as functions of Fermi level under different chemical potentials: (a) p1, (b) p2, (c) p3, and (d) p4.

5.4.4 5.4.4 DDC -- defect density and Fermi level calculations

5.4.4.1 Run DDC module

After finishing the DEC module, return to the directory ZnGeP2 and use the command `dasp 4` to execute the DDC module. No additional operation is required while waiting for the program to complete.

5.4.4.2 Workflow of DDC module

Summarize the defect-related data:

Firstly, DDC module will judge which defects have been finished calculating based on the output of DEC module, and take all these defects into consideration in the DDC calculation. Then, DDC will search for the output information about each defect as formation energy, transition energy levels, and degeneracy factor. Finally, summarize all the data and write in the file `DefectParams.txt` .

`4ddc.out` is the log file of DDC module:

```
############ Collecting information from DEC ############
Read defect types from DEC calculation successfully.
Defects considered in DDC calculation: ['P_Zn1', 'Zn_P1', 'Ge_P1', 'P_Ge1', 'P_i-3', 'P_
↪i-1', 'P_i-4', 'V_Zn1', 'Ge_Zn1', 'Zn_Ge1', 'V_P1', 'Ge_i-3', 'Ge_i-1', 'Ge_i-5', 'V_
↪Ge1', 'Zn_i-6', 'Zn_i-5', 'Zn_i-4']
Chemical potentials change from p1 to p2.
Calculate gq for defect in each charge state.
Calculate Nsites for P_Zn1: 1.245078e+22 cm^-3.
Calculate Nsites for Zn_P1: 2.490156e+22 cm^-3.
Calculate Nsites for Ge_P1: 2.490156e+22 cm^-3.
Calculate Nsites for P_Ge1: 1.245078e+22 cm^-3.
Calculate Nsites for P_i-3: 1.660104e+22 cm^-3.
Calculate Nsites for P_i-1: 1.660104e+22 cm^-3.
...
############ Collecting information from DEC ############
```

Below is the file `DefectParams.txt` :

```
1300 300
0.360000 0.540000
2.067143
P_Zn1 1.245078e+22 2 1.0959 1 0.7854 2 0.602 1 0.3478 2 1.3758 1 1.8193 2 1.8431 1 2.
↪0373 2 3.993075 2.591475
Zn_P1 2.490156e+22 2 0.9803 1 0.4186 2 0.1979 1 x x 1.3004 1 1.6611 2 1.8974 1 2.0551 2␣
↪2.013776 3.415376
Ge_P1 2.490156e+22 2 0.017 1 x x x x x x 0.6089 1 1.4605 2 1.7595 1 x x 1.214661 1.681861
P_Ge1 1.245078e+22 2 1.5928 1 0.6776 2 0.3533 1 x x 1.8788 1 2.0787 2 x x x x 2.242467 1.
↪775267
```

(continues on next page)

(continued from previous page)

```
P_i-3 1.660104e+22 2 0.463 1 0.1405 2 0.0186 1 x x 1.1848 1 x x x x x x 3.841352 3.374152
P_i-1 1.660104e+22 2 1.3134 1 0.5638 2 0.2947 1 0.112 2 1.09 1 x x x x x x 4.262836 3.
↪795636

...
```

Self-consistent calculation under growth temperature:

The DDC module will calculate the defect/dopant and carrier densities at the temperature T=1300 K, and obtain the Fermi level by self-consistently under the charge neutralization condition.

```
############ First-time self-consistent calculation ############
Fermi level at growth temperature of 1300.000000 K
Fermi level = 0.622739 eV
Fermi level = 0.626234 eV
Fermi level = 0.629768 eV
Fermi level = 0.633339 eV
Fermi level = 0.636946 eV
...
The defect density for one single defect is fixed at the value calculated at T=1300.
↪000000 K.
############ First-time self-consistent calculation ############
```

Self-consistent calculation under working (measuring) temperature:

The DDC module will recalculate the defect/dopant and carrier densities at the temperature T=300 K (*ddc_temperature = 1000 300*), and re-obtain the Fermi level by self-consistently under the charge neutralization condition.

```
############ Second-time self-consistent calculation ############
Defect densities in each charge state are redistributed.
Fermi level at working temperature of 300.000000 K
Fermi level = 0.575485 eV
Fermi level = 0.577322 eV
Fermi level = 0.579134 eV
Fermi level = 0.580921 eV
Fermi level = 0.582682 eV
...
############ Second-time self-consistent calculation ############
```

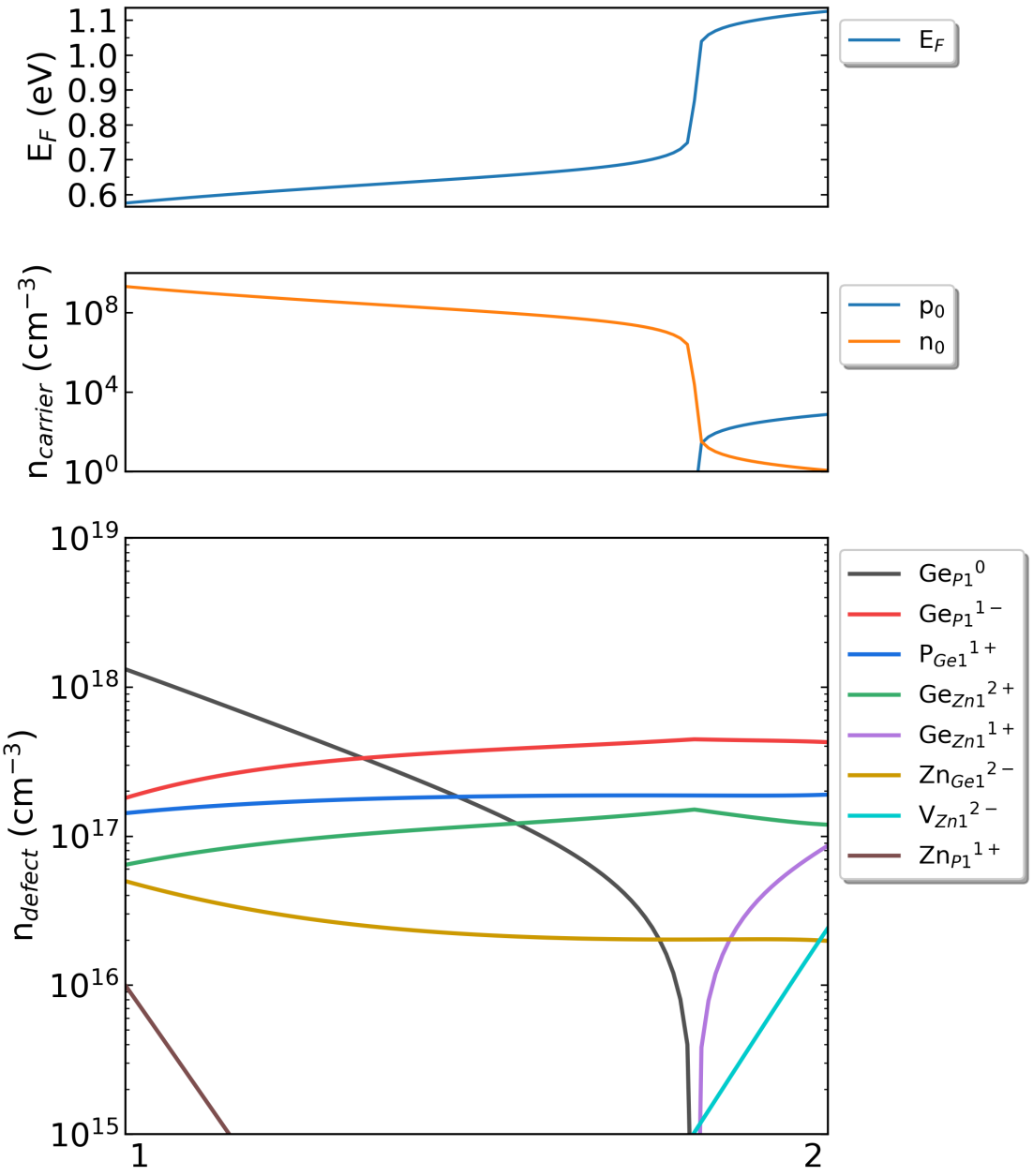
Output defect density:

DDC module will output four files in the directory ZnGeP2/ddc: `Fermi.dat` , `Carrier.dat` , `Defect_charge.dat` , and `density.png` .

```
Output Fermi level as [Fermi.dat].
Output Carrier density as [Carrier.dat].
Output Defect density as [Defect_charge.dat].
############ DDC calculation is done. ############
```

Users can plot manually based on the above data file using Origin, or refer to the image file `density.png` automatically generated by the program, where the obtained image of the change of defect density with chemical potential from p1 to p2 is shown in below:

Fig: The Fermi level, electron and hole carrier densities, and defect densities in ZnGeP2 as functions of the


1.1
1.0
0.9
0.8
0.7
0.6
E_F (eV)
E_F
10^8
10^4
10^0
$n_{carrier}$ (cm^{-3})
p_0
n_0
10^{19}
10^{18}
10^{17}
10^{16}
10^{15}
n_{defect} (cm^{-3})
$Ge_{P1}{}^{0}$
$Ge_{P1}{}^{1-}$
$P_{Ge1}{}^{1+}$
$Ge_{Zn1}{}^{2+}$
$Ge_{Zn1}{}^{1+}$
$Zn_{Ge1}{}^{2-}$
$V_{Zn1}{}^{2-}$
$Zn_{P1}{}^{1+}$
1
2

chemical potentials changed between 1 and 2. (the growth temperature is 1300 K, and the working temperature is 300 K.)

Users can adjust the growth and working temperatures in file `dasp.in` to obtain defect densities in different situations. For example, modify the parameters as follows in file `dasp.in` and run command `dasp 4` again to execute the DDC module that can obtain a new image of defect density:

```
############## DDC Module ##############
ddc_temperature = 800 300
ddc_mass = 0.36 0.54
ddc_path = 1 2
```

Fig: The Fermi level, electron and hole carrier densities, and defect densities in ZnGeP2 as functions of the chemical potentials changed between 1 and 2. (the growth temperature is 800 K, and the working temperature is 300 K.)

Or

```
############## DDC Module ##############
ddc_temperature = 300 300
ddc_mass = 0.36 0.54
ddc_path = 1 2
```

Fig: The Fermi level, electron and hole carrier densities, and defect densities in ZnGeP2 as functions of the chemical potentials changed between 1 and 2. (the growth temperature is 300 K, and the working temperature is 300 K.)

5.4.5 5.4.5 The calculations of radiative transition coefficient and photoluminescence (PL) spectrum

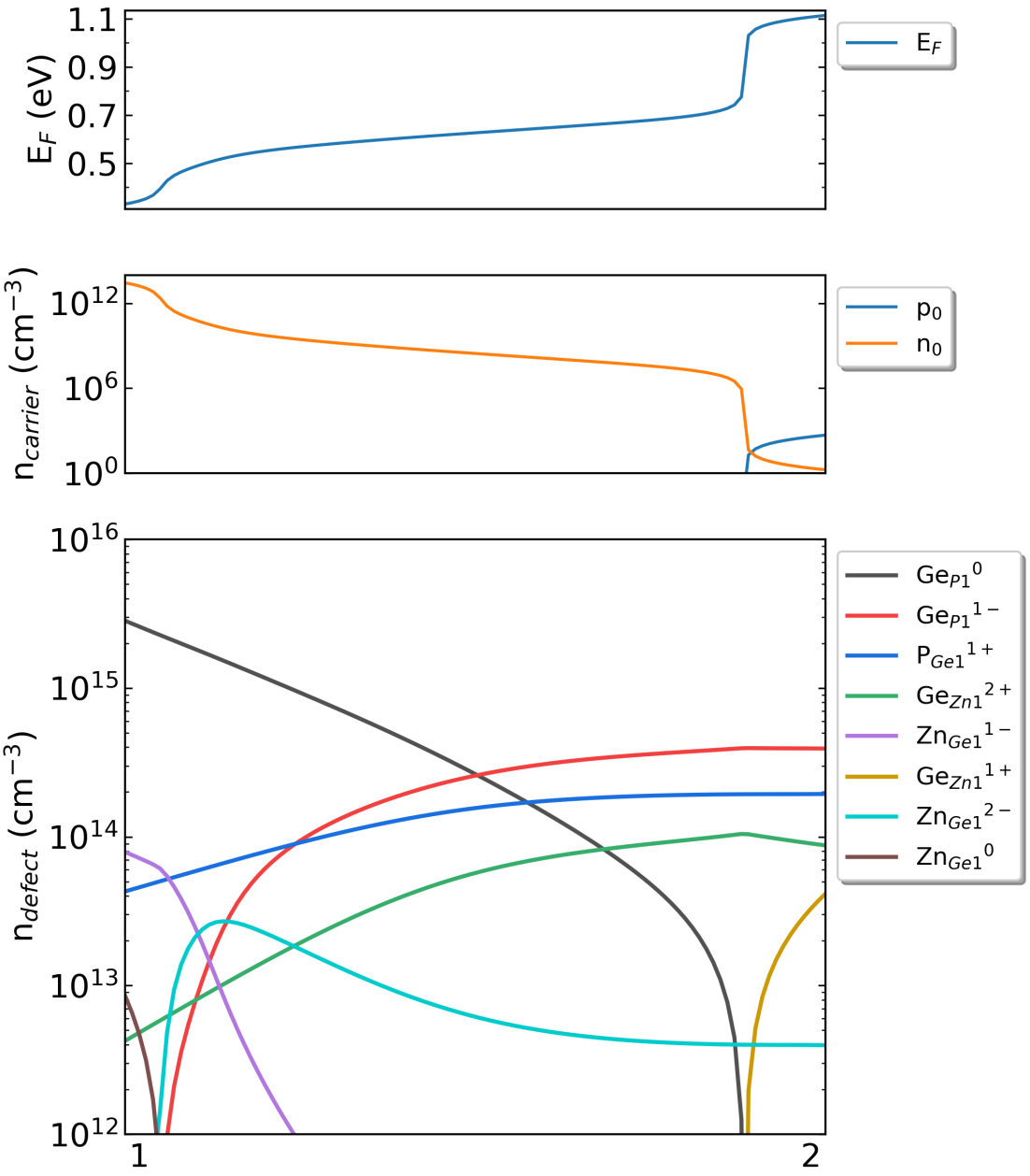
5.4.5.1 Run CDC module

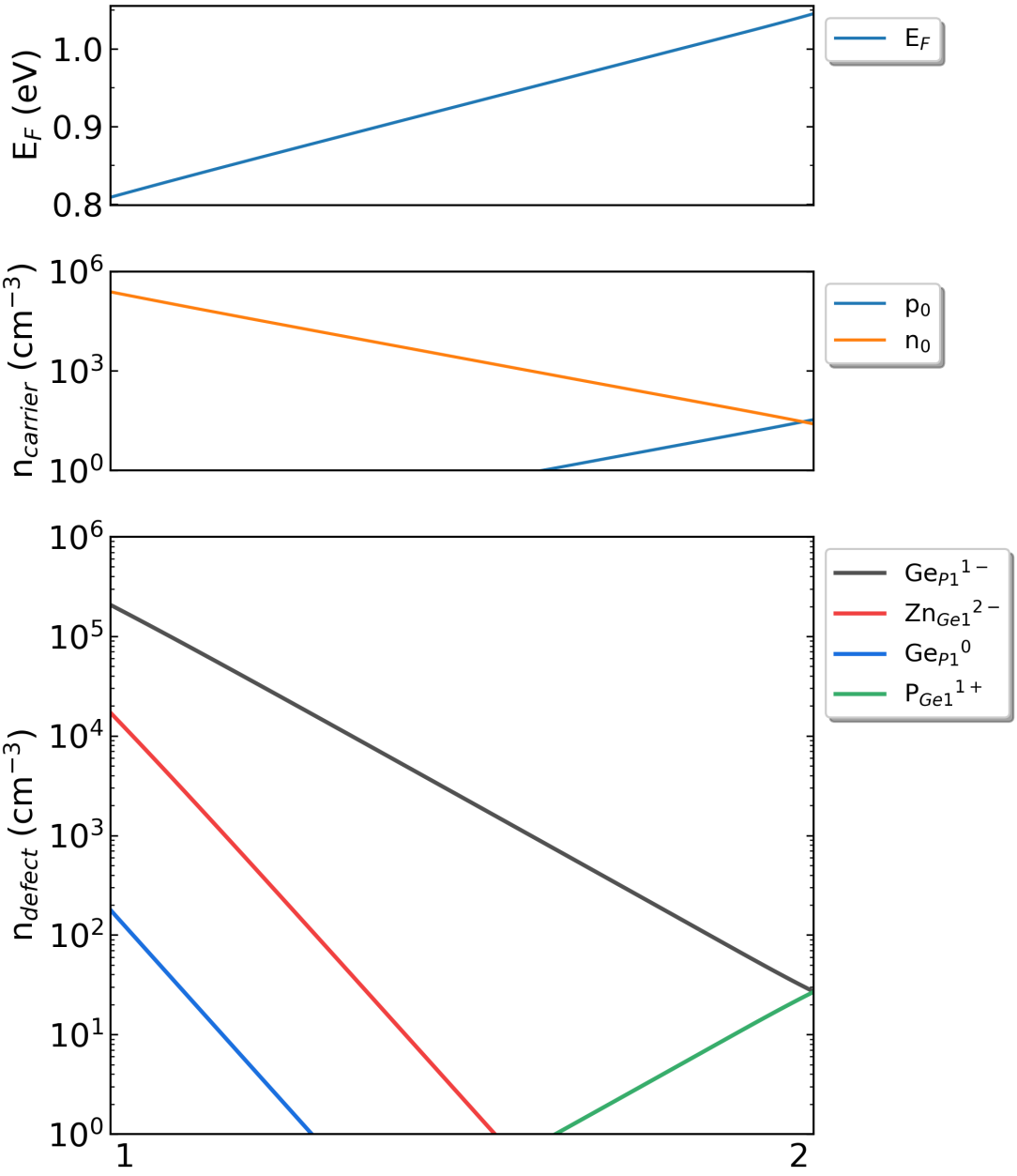
There have some things that need to be confirmed before using the CDC module:
(1) The DEC module has been completed (the DDC calculation can be skipped).
(2) For the selected defect transition process, the charge states before and after the transition should exist, and the transition energy levels should be in the band gap.
(3) If the radiative capture coefficient needs to be calculated, the VASP source program needs to be modified and recompiled, and the VASP path of the newly compiled version should be filled in *vasp_path_cdc* in file `dasp.in` .

For ZnGeP2, we want to calculate the hole trapping process of the defect Ge_{Zn} from 0 to +1 charge state, so we need to distinguish the band index of the defect level and VBM level in the defect calculation. We recommend that the band index could always be determined by the `EIGENVAL` file obtained in the static calculation of the neutral defect. For Ge_{Zn} , the band indexes for VBM and defect level are 864 and 865 respectively, and the process takes place in a spin-down channel judged by `EIGENVAL` , so the following information can be written into `dasp.in` :

```
############## CDC Module ###############
cdc_defect = Ge_Zn1
cdc_job = pl
```

(continues on next page)


1.1
0.9
0.7
0.5
E_F (eV)
E_F
10^{12}
10^{6}
10^{0}
$n_{carrier}$ (cm^{-3})
p_0
n_0
10^{16}
10^{15}
10^{14}
10^{13}
10^{12}
n_{defect} (cm^{-3})
1
2
Ge_{P1}^{0}
Ge_{P1}^{1-}
P_{Ge1}^{1+}
Ge_{Zn1}^{2+}
Zn_{Ge1}^{1-}
Ge_{Zn1}^{1+}
Zn_{Ge1}^{2-}
Zn_{Ge1}^{0}


E_F (eV)
1.0
0.9
0.8
E_F
n_carrier (cm^-3)
10^6
10^3
10^0
p_0
n_0
n_defect (cm^-3)
10^6
10^5
10^4
10^3
10^2
10^1
10^0
1
2
Ge_P1^1-
Zn_Ge1^2-
Ge_P1^0
P_Ge1^1+

(continued from previous page)

```
cdc_charge = 0 1      # The charge state before and after capture is in the front and␣
↪back respectively.
cdc_band = 864 865     # The band index for band edge and defect level is in the front␣
↪and back respectively.
cdc_temperature = 300
spin_channel = 2
refractive_index = 2.38
```

Use comman `dasp 5` to execute the CDC module, no additional operation need is required while waiting for the program to complete.

5.4.5.2 Preliminary calculation

Firstly, CDC module will judge whether the structural relaxations using HSE functional for the structure of the initial and final state of the selected defect have been completed based on the value of *level* in `dasp.in` . If *level* is 2, CDC module will adopt HSE functional to optimize the structure of the selected defect. If *level* is 3, CDC module will skip the structure optimization. And if *level* is 1, CDC module will exit the calculation of this module.

The following comes from the program log file `5cdc.out` of CDC module:

```
---------------------- relaxation calc of initial state ------------------------

finished : /data/home/.../ZnGeP2/cdc/Ge_Zn1/Radiate_calc/_q0_to_q1_/initial_state/
↪relaxation

---------------------- relaxation calc of final state ------------------------

finished : /data/home/.../ZnGeP2/cdc/Ge_Zn1/Radiate_calc/_q0_to_q1_/final_state/
↪relaxation

-------------------------------------------------
```

Next, CDC module will use the structure relaxed by HSE functional to calculate the carrier transition matrix of the initial state and the intermediate state energy of the final state with the initial structure.

The following comes from the program log file `5cdc.out` of CDC module:

```
Ge_Zn1 : from q0 state to q1 state
hole : from 864 band to 865 band

---------------------- static calc of initial state for transition matrix --------------
↪---------
```

(continues on next page)

(continued from previous page)

```
finished : /data/home/.../ZnGeP2/cdc/Ge_Zn1/Radiate_calc/_q0_to_q1_/initial_state/static_
↪optic

----------------------- static calc of intermediate state for relaxation energy ---------
↪--------------

finished : /data/home/.../ZnGeP2/cdc/Ge_Zn1/Radiate_calc/_q0_to_q1_/intermediate_state/
↪static
```

Subsequently, CDC module will calculate the zero-phonon line energy E_zpl based on the transition energy between the two defect states, and the phonon relaxation energy E_rel required by the defect from the initial structure to the final structure based on the energy of the intermediate state and final state, so as to obtain the energy of the radiated photons (E_emission).

The following comes from the program log file `5cdc.out` of CDC module:

```
transition level is 1.4203 eV
E_zpl ( Energy of zero phonon line ) is 1.4203 eV
total energy of the final state with the initial state configuration is -1142.4152 eV
total energy of the final state is -1142.6177 eV
E_rel (the lattice relaxation energy) is 0.2025 eV
E_emission (the emission energy) is 1.2178 eV
```

5.4.5.3 The flow of radiative capture coefficient calculation

After finishing the above calculations, CDC module will calculate the radiative capture coefficient according to the data such as the volume of supercell and carrier effective masses, and the corresponding equation.

The following comes from the program log file `5cdc.out` of CDC module:

```
Radiative carrier capture coefficient is 0.9106*1e-13 cm^3/s

----------------------- End of Calculation for Radiative Capture Rate --------------------
↪----
```

5.4.5.4 The flow of PL spectrum calculation

After obtaining the above data, CDC module will analyze the generalized coordinates difference Q between the initial and the final state of the defect structure relaxed by HSE functional, and linearly generate a series of structures along this direction.

There will appear many directories for static calculation in the both directories /cdc/Ge_Zn1/Radiate_calc/__q0_to_q1__/final_state and /cdc/Ge_Zn1/Radiate_calc/__q0_to_q1__/initial_state.

```
Q0  Q10  Q-10  Q2  Q-2  Q4  Q-4  Q6  Q-6  Q8  Q-8
```

After completing the above calculation, the CDC module can obtain the effective phonon energy, phonon wavefunction, and Huang-Rhys factor of the final state in the initial and final state according to the generated structure and the corresponding defect formation energy, and also obtain the one-dimensional configuration diagram of the initial and final state of the defect. The output is the image `ccdiagram.png` , as shown in below:

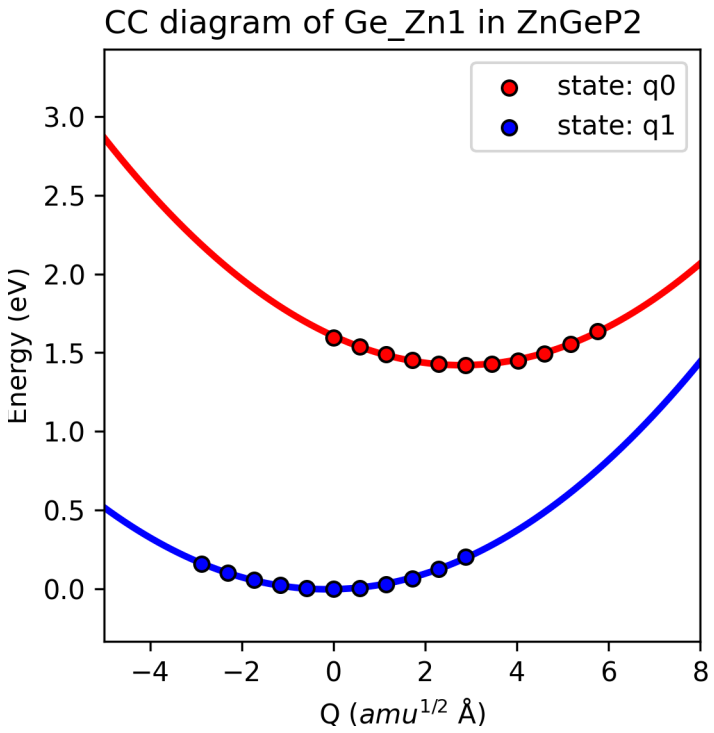


Fig: The one-dimensional configuration diagram of defect Ge_Zn1 in ZnGeP2.

The following comes from the program log file `5cdc.out` of CDC module:

```
---------------------- Calculation for PL Spectrum Start ------------------------

Analysing deltaQ (the structure difference in generalized coordinate) ...
deltaQ between two structures in a.u.:232.2316
```

(continues on next page)

(continued from previous page)

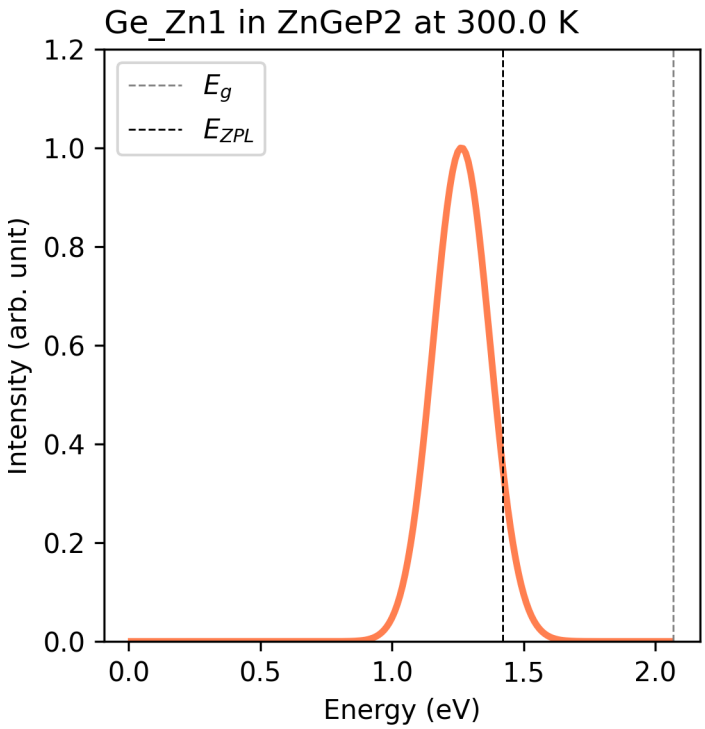
```
deltaQ between two structures in amu^1/2*Angs: 2.879

------------------------------------------------

Generating structures...

...

------------------------------------------------

calculation of initial state and final state all finished

------------------------------------------------

analysing for pl spectrum...

effective phonon energy of the initial state is 0.01397 eV

effective phonon energy of the final state is 0.01338 eV

Huang-Rhys factor of the final state is 15.13453

ccdiagram.png saved in dir /data/home/Zn_Ge_P/cdc/Ge_Zn1/Radiate_calc/_q0_to_q1_
```

Finally, CDC module will output the raw data file `lineshape.dat` of the PL spectrum, and also output its peak position and FWHM, at last, save the image in file `lineshape.png` .

```
lineshape.dat saved in dir /data/home/cai/daike/Zn_Ge_P/CDC_test_0310/cdc/Ge_Zn1/Radiate_
↪calc/_q0_to_q1_

Position of the peak in the lineshape is 1.26 eV

Full width at half maxima of the lineshape is 0.21 eV

analysis for pl spectrum finished

lineshape.png saved in dir /data/home/Zn_Ge_P//cdc/Ge_Zn1/Radiate_calc/_q0_to_q1_

----------------------- End of Calculation for PL Spectrum -----------------------
```

The image `lineshape.png` is shown in below:

Fig: The PL spectrum of defect Ge_Zn1 in ZnGeP2 at a given temperature.



5.5 5.5 Rapid prediction of the stability of double perovskite materials

In the above cases, we showed that the TSC module can calculate the element chemical potential and uses it to calculate the formation energy in DEC module. In addition, TSC module can be run independently to analyze the stability of the compound without the need for PREPARE module calculations.

The TSC module can be run directly by setting *tsc_only* = *T* in `dasp.in` . After TSC module finishes the first stage analysis, it will conduct the second stage analysis with *level* = *1* by itself.

The following examples showed the analysis process of three kinds of double perovskite materials Cs2AgBiCl6, Rb2LiInI, and K2LiYF6.

5.5.1 5.5.1 Cs2AgBiCl6 (Predicted result: stable)

5.5.1.1 Prepare files

The first step in the material stability analysis using TSC module is still to prepare `POSCAR` and `dasp.in` .

The `POSCAR` of Cs2AgBiCl6 can be obtained from MP database, and the structure needs to be optimized or set by the user. The `POSCAR` for this case is as follows:

```
Cs2 Ag1 Bi1 Cl6
1.00000000000000
7.7438184481880610    0.0000000000000355    0.0000000000000251
3.8719092240440918    6.7063434983622878   -0.0000000000000092
3.8719092240440918    2.2354478328207339    6.3228012862560394
Cs   Ag   Bi   Cl
```

(continues on next page)

(continued from previous page)

```
2     1     1     6
Direct
0.7500000000000000  0.7500000000000000  0.7500000000000000
0.2500000000000000  0.2500000000000000  0.2500000000000000
0.5000000000000000  0.5000000000000000  0.5000000000000000
-0.0000000000000000 -0.0000000000000000  0.0000000000000000
0.7508700137251050  0.2491299562748926  0.2491299562748926
0.2491299562748926  0.2491299562748926  0.7508700137251050
0.2491299562748926  0.7508700137251050  0.7508700137251050
0.2491299562748926  0.7508700137251050  0.2491299562748926
0.7508700137251050  0.2491299562748926  0.7508700137251050
0.7508700137251050  0.7508700137251050  0.2491299562748926
```

Users need to set the relevant parameters in the file `dasp.in` according to their own situation, and set *tsc_only* = *T* and *database_api* .

```
############### Job Scheduling ###############
cluster = SLURM     # (job scheduling system)
node_number = 4          # (number of node)
core_per_node = 32       # (core per node)
queue = normal            # (name of queue/partition)
max_time = 24:00:00         # (maximum time for a single DFT calculation)
vasp_path_tsc = /opt/vasp.5.4.4/bin/vasp_std
job_name = submit_job     # (name of script)
potcar_path = /opt/POT/potpaw_PBE     # (path of pseudopotentials)
max_job = 5

############### TSC Module ###############
database_api = ******************** # (str-list type)
tsc_only = T
plot_3d = T
```

For the parameters of TSC module,

```
database_api = ******************** # (str-list type)
# use to visit the Materials Project database
```

```
tsc_only = T
# Only rapid stability analysis with level = 1 is performed.
```

```
plot_3d = T
# For quaternary compounds, TSC module can output three dimensional phase diagram (for␣
↪user reference only). The default is F, it will be output three dimensional phase␣
↪diagram by setting to T.
```

5.5.1.2 Calculation and analysis

The total energy calculation of the host structure (the parameters are consistent with MP database):

TSC module will use the same input parameters (INCAR, KPOINTS, POTCAR) with the `Materials Project` database to perform structural relaxation and static calculation on the primitive cells given by the user. Therefore, the calculated total energy is comparable to that of the MP database. This step is to obtain the **key hetero-phases** that limit the stability of Cs2AgBiCl6. In the directory, we can see:

```
cd tsc
cd Cs2AgBiCl6/
ls
relaxation1  relaxation2  static
```

The running log also can be seen from the Cs2AgBiCl6/tsc/2tsc.out, that is, the steps such as generating input files, relaxation1, relaxation2, static and data extraction.

Rapid analysis of the stability and key hetero-phases compounds:

The TSC module will search for all the secondary compounds that compete with Cs2AgBiCl6 in the MP database. According to the output file `materials_info.yaml` , it can be found that all the considered hetero-phases compounds including:

```
secondary_phases:
- - Cs
  - Ag
  - Bi
  - Cl2
  - Ag3Bi
  - Ag2Cl3
  - Ag3Cl
  - AgCl
  - Cs2AgCl3
  - CsAgCl2
  - CsAgCl3
  - Bi6Cl7
  - BiCl3
  - Cs3BiCl6
  - Cs3Bi2Cl9
  - CsBi
  - Cs3Bi2
  - CsBi2
  - Cs3Bi
  - CsCl
```

By comparing the calculated total energy of Cs2AgBiCl6 with that of the hetero-phases extracted from the database, it can be judged that Cs2AgBiCl6 is **thermodynamically stable**.

Subsequently, the program will automatically download the key hetero-phases compounds that can limit the thermodynamic stability of Cs2AgBiCl6. Ag, Cs, Bi, Cl2, AgCl, CsAgCl2, Cs3BiCl6, CsAgCl3, and Cl2Cs3Bi2Cl9 are considered in this case. The relevant information can be seen in `2tsc.out` :

```
...
analysing the thermodynamic stability of Cs2AgBiCl6.
The stability of Cs2AgBiCl6 is: True.
key phases of Cs2AgBiCl6 are: AgCl Ag Cs3Bi2Cl9 CsAgCl2 Cs3BiCl6 CsAgCl3 Cs Bi Cl2 .
```

(continues on next page)

(continued from previous page)

```
analysing of Cs2AgBiCl6 is done.
...
```

Meanwhile, the same information is also output in file `materials_info.yaml` :

```
key_phases:
  - AgCl
  - CsAgCl2
  - CsAgCl3
  - Ag
  - Cs3BiCl6
  - Cs3Bi2Cl9
  - Cs
  - Bi
  - Cl2
```

The total energy calculation of the host and hetero-phase compounds:

As Cs2AgBiCl6 is thermodynamically stable, TSC module will perform the second analysis and calculation after the key hetero-phase compounds are confirmed, i.e. according to the POTCAR files in the path of pseudopotential provided by users, using *level=1* to calculate the total energies of Cs2AgBiCl6 and the hetero-phase compounds mentioned above. `2tsc.out` is as follows:

```
...
Job ******** submitted: /home/test/Cs2AgBiCl6/tsc/Cs2AgBiCl6/static_recalc
Job ******** submitted: /home/test/Cs2AgBiCl6/tsc/AgCl/static_recalc
Job ******** submitted: /home/test/Cs2AgBiCl6/tsc/CsAgCl2/static_recalc
...
Succeed job  12267.host2: /home/test/Cs2AgBiCl6/tsc/Cs2AgBiCl6/static_recalc
Succeed job  12269.host2: /home/test/Cs2AgBiCl6/tsc/AgCl/static_recalc
Succeed job  12271.host2: /home/test/Cs2AgBiCl6/tsc/CsAgCl2/static_recalc
...
```

If there are errors in the calculations, modify the relevant parameters, such as `INCAR` or `KPOINTS` , and then run again. The details can be seen in Common problems section.

The chemical potential calculation:

Calculating the formation energy and stable chemical potential region of Cs2AgBiCl6 based on the calculated total energy. TSC module will give the endpoint of eight chemical potentials, and write them into `dasp.in` :

```
# The orders are consistent with the order of elements in POSCAR, i.e. the first column␣
↪is Cs, the second column is Ag, the third column is Bi, and the fourth is Cl.
E_pure = -0.836 -2.6987 -3.8871 -1.7877
p1 = -3.5853 -0.4806 -2.9502 -0.5666
p2 = -3.1047 -0.0 -1.5084 -1.0472
p3 = -4.1861 -0.781 -3.2506 -0.2662
p4 = -3.4051 -0.0 -0.9076 -1.0472
p5 = -3.5373 -0.5286 -2.9982 -0.5666
p6 = -3.0087 0.0 -1.4124 -1.0952
p7 = -3.7897 -0.781 -3.2506 -0.3983
p8 = -3.0087 -0.0 -0.9076 -1.1793
```

The output after the program is completed can be seen in `2tsc.out` :

```
analysing the thermodynamic stability of Cs2AgBiCl6.
The stability of Cs2AgBiCl6 is: True.
key phases of Cs2AgBiCl6 are: AgCl Ag Cs3Bi2Cl9 CsAgCl2 Cs3BiCl6 CsAgCl3 Cs Bi Cl2 .
analysing of Cs2AgBiCl6 is done.
sub-module of tsc: 'auto thermodynamic calculation' ends successfully.
----------------------------
DASP-TSC finished
```

For the ternary and multinary compounds, TSC module will output the image of the stable region and the chemical potential at the endpoint of the stable region. For the two dimensional figures, it can be seen through this directory:

```
cd tsc
cd 2d-figures/
ls
fig-Cs2AgBiCl6.png  fig-Cs2AgBiCl6_recalc.png  stable_2d.out stable_recalc_2d.out
```

The four files under the directory Cs2AgBiCl6/tsc/2d-figures/ are the images of the stable region plotted during the two calculations and analyses, and the chemical potential at each endpoint in the images.

Check out the files `stable_2d.out` and `fig-Cs2AgBiCl6.png` . The horizontal and vertical coordinates of `fig-Cs2AgBiCl6.png` are the chemical potential of the elements marked in the figure, and the shaded area is the stable region of the host compound. Each line on the boundary is the chemical potential curve at the critical condition where the marked material can or can not form. This is the image output from the first calculation and analysis process.

```
     Bi       Ag       Cs       Cl
-------  -------  -------  -------
-2.9709  -0.3807  -3.5994  -0.5224
-3.1086  -0.5184  -3.5994  -0.4765
-2.8656  -0.5184  -3.5994  -0.517
-1.6748  -0.1214  -3.5994  -0.7817
```

Fig: The stable chemical potential region of Cs2AgBiCl6. (comes from the MP database)

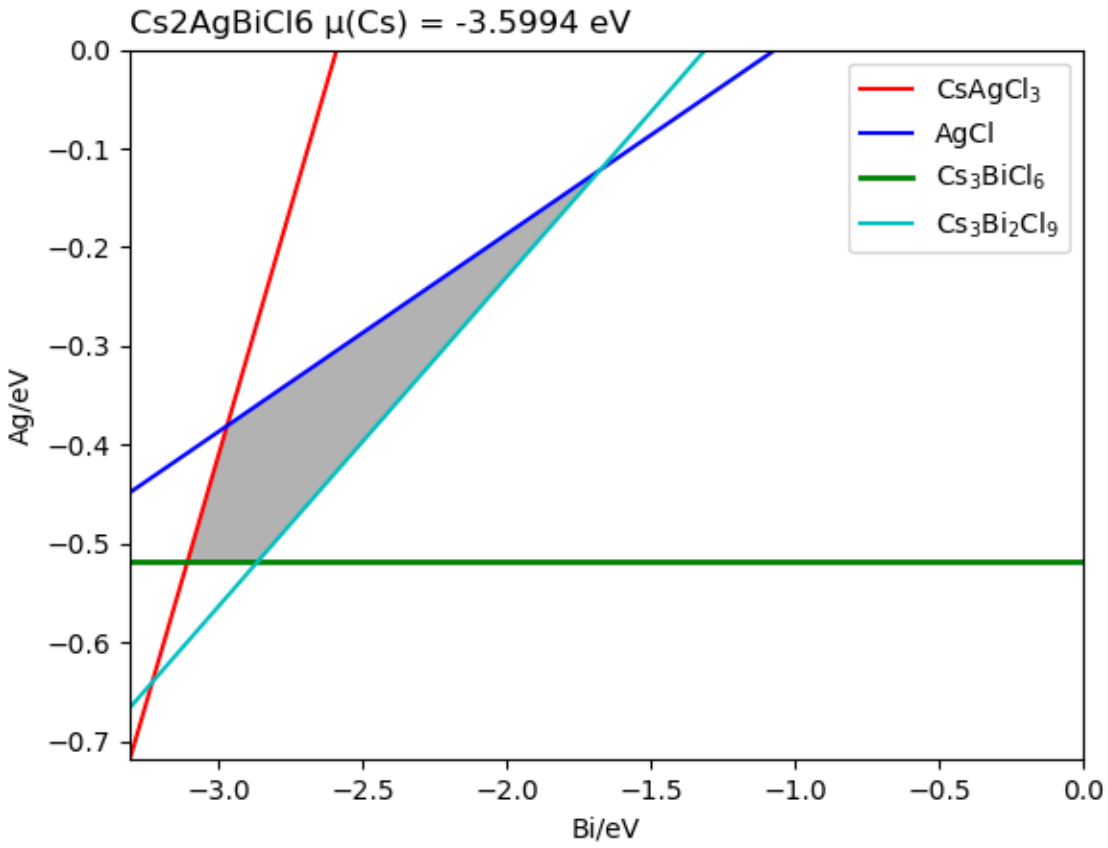
Check out the files `stable_recalc_2d.out` and `fig-Cs2AgBiCl6_recalc.png` . They are the image and data output from the second calculation and analysis process.

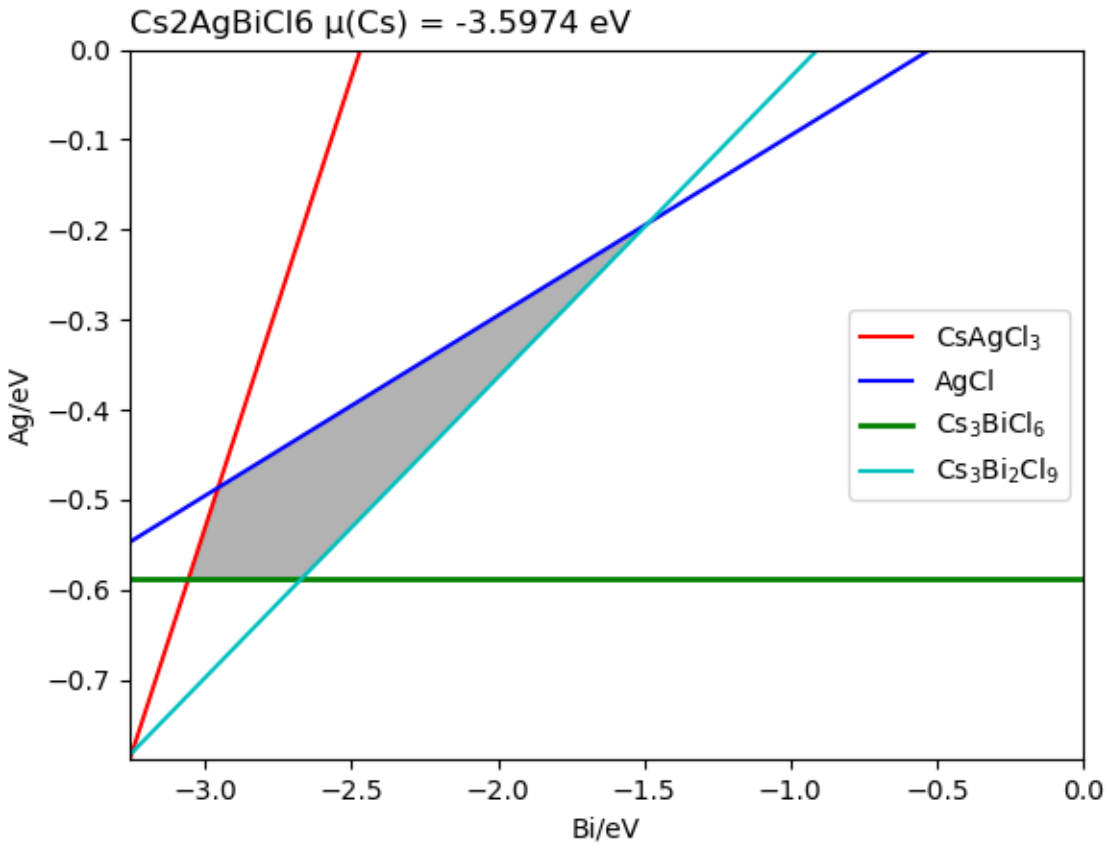
```
     Bi       Ag       Cs       Cl
-------  -------  -------  -------
-2.9562  -0.4866  -3.5974  -0.5606
-3.0583  -0.5887  -3.5974  -0.5265
-2.6737  -0.5887  -3.5974  -0.5906
-1.4845  -0.1923  -3.5974  -0.8549
```

Fig: The stable chemical potential region of Cs2AgBiCl6. (comes from the second stage calculations)

For the three dimensional figures, it can be seen through this directory:

```
cd tsc
cd 3d-figures/
ls
fig-Cs2AgBiCl6_3d.png  fig-Cs2AgBiCl6_3d_recalc.png  stable.out  stable_recalc.out
```


Cs2AgBiCl6 μ(Cs) = -3.5994 eV
CsAgCl3
AgCl
Cs3BiCl6
Cs3Bi2Cl9
Ag/eV
Bi/eV
0.0
−0.1
−0.2
−0.3
−0.4
−0.5
−0.6
−0.7
−3.0
−2.5
−2.0
−1.5
−1.0
−0.5
0.0


Cs2AgBiCl6 μ(Cs) = -3.5974 eV
0.0
−0.1
−0.2
−0.3
−0.4
−0.5
−0.6
−0.7
Ag/eV
−3.0
−2.5
−2.0
−1.5
−1.0
−0.5
0.0
Bi/eV
CsAgCl3
AgCl
Cs3BiCl6
Cs3Bi2Cl9

The four files under the directory Cs2AgBiCl6/tsc/3d-figures/ are the images of the stable region plotted during the two calculations and analyses, and the chemical potential at each endpoint in the images.

Check out the files `stable.out` and `fig-Cs2AgBiCl6_3d.png` . The chemical potentials of the three elements marked in `fig-Cs2AgBiCl6_3d.png` constitute three coordinate axes, and the area surrounded by red lines is the three-dimensional stable region of Cs2AgBiCl6. This is the image output from the first calculation and analysis process, and the coordinates of each point in this region can be obtained from the file `stable.out` .

Cs	Ag	Bi	Cl
-3.17837	6.73289e-17	-1.90967	-0.903106
-3.47796	-0	-1.31049	-0.903106
-3.08102	-3.47289e-18	-1.81231	-0.951782
-3.08102	-0	-1.31049	-1.03542
-3.51867	-0.340294	-2.93055	-0.562812
-4.11785	-0.639883	-3.23014	-0.263223
-3.46999	-0.38897	-2.97922	-0.562812
-3.7209	-0.639883	-3.23014	-0.395537

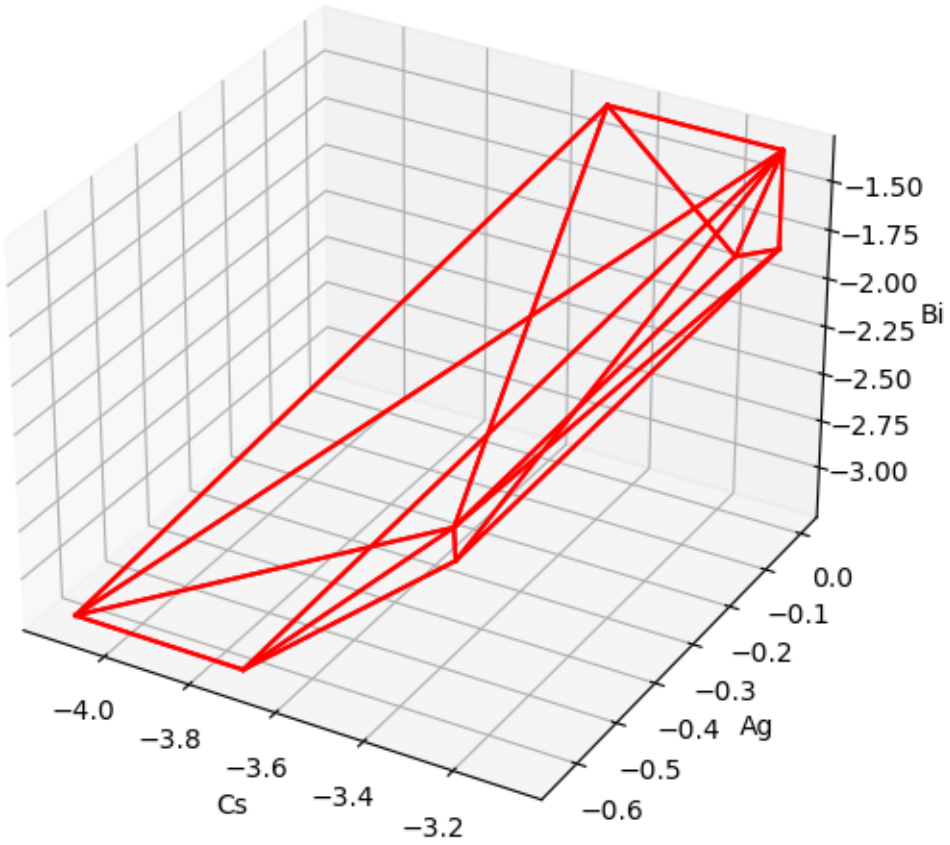


Fig: The 3D stable chemical potential region of Cs2AgBiCl6. (comes from the MP database)

Check out the files `stable_recalc_3d.out` and `fig-Cs2AgBiCl6_3d_recalc.png` . They are the image and data output from the second calculation and analysis process.

Cs	Ag	Bi	Cl
-3.5853	-0.4806	-2.9502	-0.5666
-3.1047	-3.63857e-17	-1.5084	-1.0472
-4.1861	-0.781	-3.2506	-0.2662
-3.4051	-0	-0.9076	-1.0472
-3.5373	-0.5286	-2.9982	-0.5666
-3.0087	0	-1.4124	-1.0952
-3.7897	-0.781	-3.2506	-0.398333
-3.0087	-0	-0.9076	-1.17933

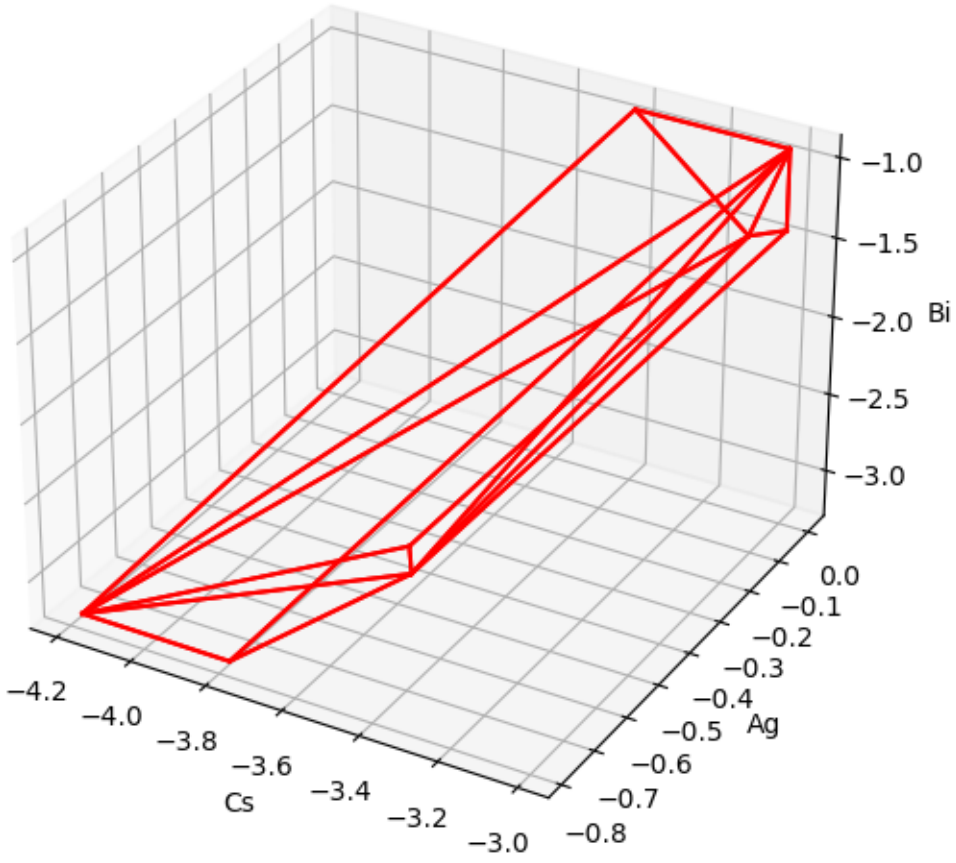


Fig: The 3D stable chemical potential region of Cs2AgBiCl6. (comes from the second stage calculations)

5.5.2 5.5.2 K2LiYF6 (Predicted result: unstable)

5.5.2.1 Prepare POSCAR and dasp.in

The `POSCAR` of K2LiYF6 can be obtained from MP database, and the structure needs to be optimized or set by the user. The `POSCAR` for this case is as follows:

```
K8 Li4 Y4 F24
1.0
8.557390 0.000000 0.000000
0.000000 8.557390 0.000000
0.000000 0.000000 8.557390
K Li Y F
8 4 4 24
direct
0.250000 0.250000 0.750000 K
0.250000 0.750000 0.750000 K
0.250000 0.750000 0.250000 K
0.250000 0.250000 0.250000 K
0.750000 0.250000 0.250000 K
0.750000 0.750000 0.250000 K
0.750000 0.750000 0.750000 K
0.750000 0.250000 0.750000 K
0.500000 0.000000 0.000000 Li
0.500000 0.500000 0.500000 Li
0.000000 0.000000 0.500000 Li
0.000000 0.500000 0.000000 Li
...
```

Fig: The crystal structure schematic of K2LiYF6.

Users need to set the relevant parameters in the file `dasp.in` according to their own situation, and set *tsc_only* = *T* and *database_api* . The details can refer to Cs2AgBiCl6.

5.5.2.2 Calculation and analysis

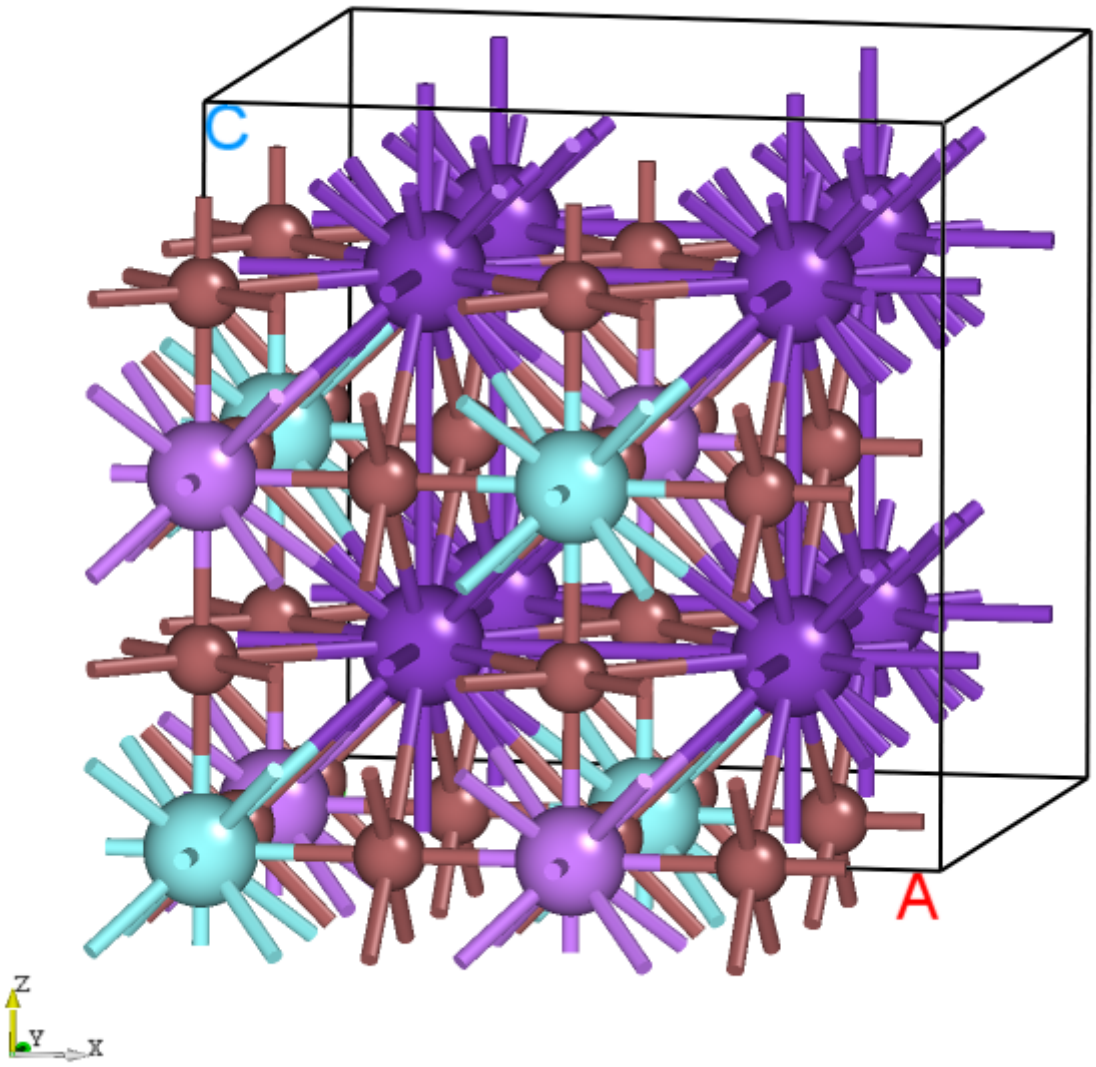
The total energy calculation of the host structure (the parameters are consistent with MP database):

TSC module will use the same input parameters (INCAR, KPOINTS, POTCAR) with the `Materials Project` database to perform structural relaxation and static calculation on the primitive cells given by the user. Therefore, the calculated total energy is comparable to that of the MP database. This step is to obtain the **key hetero-phases** that limit the stability of K2LiYF6. In the directory, we can see:

```
cd tsc
cd K2LiYF6/
ls
relaxation1  relaxation2  static
```

The running log also can be seen from the K2LiYF6/tsc/2tsc.out, that is, the steps such as generating input files, relaxation1, relaxation2, static and data extraction.

Rapid analysis of the stability and key hetero-phases compounds:


C
A
Z
Y
X

The TSC module will search for all the secondary compounds that compete with K2LiYF6 in the MP database. According to the output file `materials_info.yaml` , it can be found that all the considered hetero-phases compounds including:

```
secondary_phases:
- - K
  - Li
  - Y
  - F
  - KF2
  - KF3
  - KF
  - KF5
  - KLiYF5
  - K3YF6
  - KYF4
  - K2YF5
  - KY3F10
  - KY2F7
  - LiF
  - LiF3
  - LiYF4
  - LiYF2
  - Li3YF6
  - YF3
  - K3Li
  - KLi3
  - K3Y
  - LiY3
  - Li3Y
```

By comparing the calculated total energy of Cs2AgBiCl6 with that of the hetero-phases extracted from the database, it can be judged that Cs2AgBiCl6 is **unstable**. The relevant information can be seen in `2tsc.out` :

```
...
analysing the thermodynamic stability of K2LiYF6.
The stability of K2LiYF6 is: False.
K2LiYF6 may decompose into ['K2YF5', 'K2YF5'].
you can set tag: 'excluded_phase' to get some reference values of chemical potentials.
analysing of K2LiYF6 is done.
...
```

The value of energy above hull of K2LiYF6 is output in file `materials_info.yaml` , it is positive and consistent with the result that "this compound is unstable" .

```
e_above_hull: 0.0466
```

The decomposed products include K2YF5 and LiF, and this also can be known from the output file `materials_info.yaml` :

```
decomp:
- K2YF5
- LiF
```

As K2LiYF6 is unstable, TSC module can not judge the key hetero-phase compounds or output the images of the stable region. Moreover, TSC module will not do the calculation of the second stage further.

If users still want to do the second stage calculations to obtain the stable chemical potential region, they can set *excluded_phase* to exclude the effect of some hetero-phase compounds on the stability of the host compound based on the prompt in file `2tsc.out` . Generally, one or more of the decompositions is excluded to make the host compound has a referable stable region. i.e. set *excluded_phase = LiF* , or *excluded_phase = K2YF5* , or *excluded_phase = LiF K2YF5* in file `dasp.in` , and run TSC module again. At this point, it is possible that the compound is still unstable, then some compounds in the current decomposition path can be appended to *excluded_phase* and re-run the TSC module.

5.5.2.3 Additional calculation and analysis

In this case, the parameters in file `dasp.in` that may make the host compound has a stable region after excluding some certain hetero-phase compounds are as follows:

```
############### Job Scheduling ###############
cluster = SLURM     # (job scheduling system)
node_number = 4          # (number of node)
core_per_node = 32       # (core per node)
queue = normal            # (name of queue/partition)
max_time = 24:00:00         # (maximum time for a single DFT calculation)
vasp_path_tsc = /opt/vasp.5.4.4/bin/vasp_std
job_name = submit_job    # (name of script)
potcar_path = /opt/POT/potpaw_PBE    # (path of pseudopotentials)
max_job = 5

############### TSC Module ###############
database_api = ******************** # (str-list type)
tsc_only = T
plot_3d = T
excluded_phase = K2LiYF6 K2YF5 LiF KLiYF5 LiYF4
```

The excluded phases include K2LiYF6, K2YF5, LiF, KLiYF5, and LiYF4.

For the unstable compounds, the stable region may be different because the hetero-phases excluded are different, the above parameters are for reference only.

As the total energy calculations have already been done, so the relevant stability analysis information can be seen in file `2tsc.out` quickly:

```
analysing the thermodynamic stability of K2LiYF6.
excluded phase of K2LiYF6: K2LiYF6 K2YF5 LiF KLiYF5 LiYF4 .
The stability of K2LiYF6 is: True.
key phases of K2LiYF6 are: KYF4 K3YF6 Li3YF6 KF2 KF Li KY3F10 KY2F7 F2 K Y .
file key_phases_info_recalc.yaml generated.
analysing of K2LiYF6 is done.
```

And the second stage calculations have begun for the key hetero-phase compounds mentioned above, shown in `2tsc.out` as below:

```
...
Job ******** submitted: /home/fudan/daike/KLiYF/DASP_0219/tsc/K2LiYF6/static_recalc
Job ******** submitted: /home/fudan/daike/KLiYF/DASP_0219/tsc/KYF4/static_recalc
```

(continues on next page)

(continued from previous page)

```
Job ******** submitted: /home/fudan/daike/KLiYF/DASP_0219/tsc/K3YF6/static_recalc
Job ******** submitted: /home/fudan/daike/KLiYF/DASP_0219/tsc/Li3YF6/static_recalc
...
Succeed job  ********: /home/fudan/daike/KLiYF/DASP_0219/tsc/K2LiYF6/static_recalc
Succeed job  ********: /home/fudan/daike/KLiYF/DASP_0219/tsc/KYF4/static_recalc
Succeed job  ********: /home/fudan/daike/KLiYF/DASP_0219/tsc/K3YF6/static_recalc
Succeed job  ********: /home/fudan/daike/KLiYF/DASP_0219/tsc/KF2/static_recalc

...
```

If there are errors in the calculations, modify the relevant parameters, such as `INCAR` or `KPOINTS` , and then run again. The details can be seen in Common problems section.

The chemical potential calculation:

Calculating the formation energy and stable chemical potential region of K2LiYF6 based on the calculated total energy. TSC module will give the endpoint of 14 chemical potentials, and write them into `dasp.in` :

```
# The orders are consistent with the order of elements in POSCAR, i.e. the first column␣
↪is K, the second column is Li, the third column is Y, and the fourth is F.
E_pure = -1.086 -1.8579 -6.453 -1.8583
p1 = -0.0059 0.0 -0.1727 -5.5633
p2 = -5.5692 -5.5633 -16.8626 0.0
p3 = -0.4569 0.0 -0.6237 -5.3378
p4 = -5.7947 -5.3378 -16.6371 0.0
p5 = -5.4204 -5.4145 -16.9064 -0.0671
p6 = -5.5546 -5.5487 -16.9064 0.0
p7 = -0.0059 0.0 -0.6629 -5.4816
p8 = -5.4204 -4.1335 -18.1874 -0.0671
p9 = -5.5546 -4.2677 -18.1874 0.0
p10 = -1.2869 0.0 -5.7869 -4.2006
p11 = -1.2869 0.0 -2.0624 -4.8213
p12 = -6.1083 -4.8213 -16.5265 0.0
p13 = -0.5489 0.0 -0.7709 -5.2826
p14 = -5.8315 -5.2826 -16.6187 0.0
```

The output after the program is completed can be seen in `2tsc.out` :

```
analysing the thermodynamic stability of K2LiYF6.
excluded phase of K2LiYF6: K2LiYF6 K2YF5 LiF KLiYF5 LiYF4 .
The stability of K2LiYF6 is: True.
key phases of K2LiYF6 are: K3YF6 Li3YF6 KF2 KF KYF4 F2 Li KY3F10 KY2F7 K Y .
analysing of K2LiYF6 is done.
sub-module of tsc: 'auto thermodynamic calculation' ends successfully.
----------------------------
DASP-TSC finished
```

For the ternary and multinary compounds, TSC module will output the image of the stable region and the chemical potential at the endpoint of the stable region. For the two dimensional figures, it can be seen through this directory:

```
cd tsc
cd 2d-figures/
```

(continues on next page)

(continued from previous page)

```
ls
fig-K2LiYF6.png  fig-K2LiYF6_recalc.png  stable_2d.out stable_recalc_2d.out
```

The four files under the directory K2LiYF6/tsc/2d-figures/ are the images of the stable region plotted during the two calculations and analyses, and the chemical potential at each endpoint in the images.

Check out the files `stable_2d.out` and `fig-K2LiYF6.png` . The horizontal and vertical coordinates of `fig-K2LiYF6.png` are the chemical potential of the elements marked in the figure, and the shaded area is the stable region of the host compound. Each line on the boundary is the chemical potential curve at the critical condition where the marked material can or can not form. This is the image output from the first calculation and analysis process.

```
       Y       Li        K        F
--------  -------  -------  -------
-7.3952  -1.7304  -3.0425  -3.0094
-11.1139  -1.7304  -3.0425  -2.3896
-9.8349  -3.0094  -3.0425  -2.3896
-9.3452  -3.0094  -3.0425  -2.4712
-8.4446  -2.5591  -3.0425  -2.6964
-8.3183  -2.4689  -3.0425  -2.7325
```

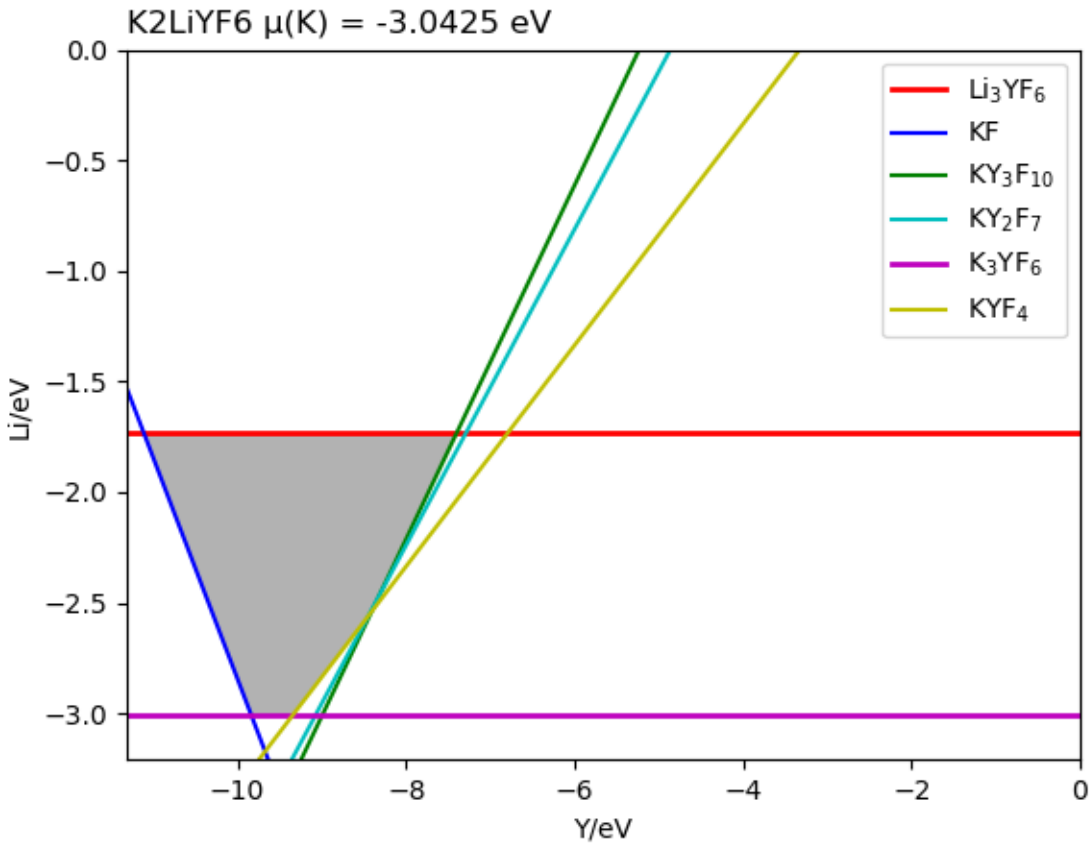


Fig: The stable chemical potential region of K2LiYF6. (comes from the MP database)

Check out the files `stable_recalc_2d.out` and `fig-K2LiYF6_recalc.png` . They are the image and data

output from the second calculation and analysis process.

```
     Y        Li         K         F
--------  -------  -------  -------
-7.373   -1.7702  -3.0571  -3.0512
-11.0975  -1.7702  -3.0571  -2.4304
-9.8165  -3.0512  -3.0571  -2.4304
-9.3263  -3.0512  -3.0571  -2.5121
-8.4243  -2.6002  -3.0571  -2.7376
-8.2955  -2.5082  -3.0571  -2.7744
```

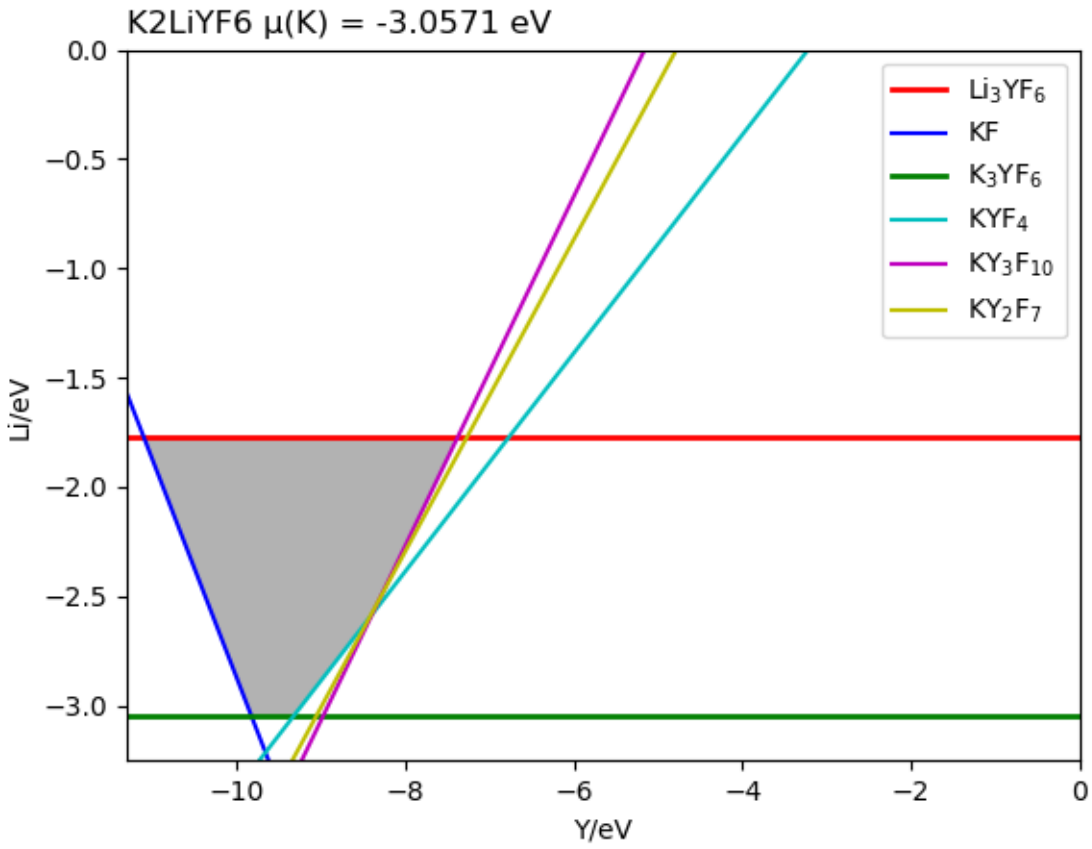


Fig: The stable chemical potential region of K2LiYF6. (comes from the second stage calculations)

For the three dimensional figures, it can be seen through this directory:

```
cd tsc
cd 3d-figures/
ls
fig-K2LiYF6_3d.png  fig-K2LiYF6_3d_recalc.png  stable.out  stable_recalc.out
```

The four files under the directory K2LiYF6/tsc/3d-figures/ are the images of the stable region plotted during the two calculations and analyses, and the chemical potential at each endpoint in the images.

Check out the files `stable.out` and `fig-K2LiYF6_3d.png` . The chemical potentials of the three elements marked in `fig-K2LiYF6_3d.png` constitute three coordinate axes, and the area surrounded by red lines is the three-dimensional stable region of K2LiYF6. This is the image output from the first calculation and analysis process, and the coordinates of each point in this region can be obtained from the file `stable.out` .

K	Li	Y	F
-0.0331078	0	-0.806716	-5.39901
-1.31209	0	-5.92265	-4.12003
-0.0331078	0	-0.316988	-5.48063
-0.483403	0	-0.767283	-5.25549
-0.573574	0	-0.911557	-5.20138
-1.31209	2.75724e-17	-2.20396	-4.73981
-5.49866	-5.46555	-16.8041	5.92119e-16
-5.49866	-4.18657	-18.0831	5.92119e-16
-5.51374	-5.48063	-16.7589	1.77636e-15
-5.73889	-5.25549	-16.5337	1.77636e-15
-5.77496	-5.20138	-16.5157	1.18424e-15
-6.0519	-4.73981	-16.4234	0
-5.36558	-5.33247	-16.8041	-0.0665391
-5.36558	-4.05349	-18.0831	-0.0665391

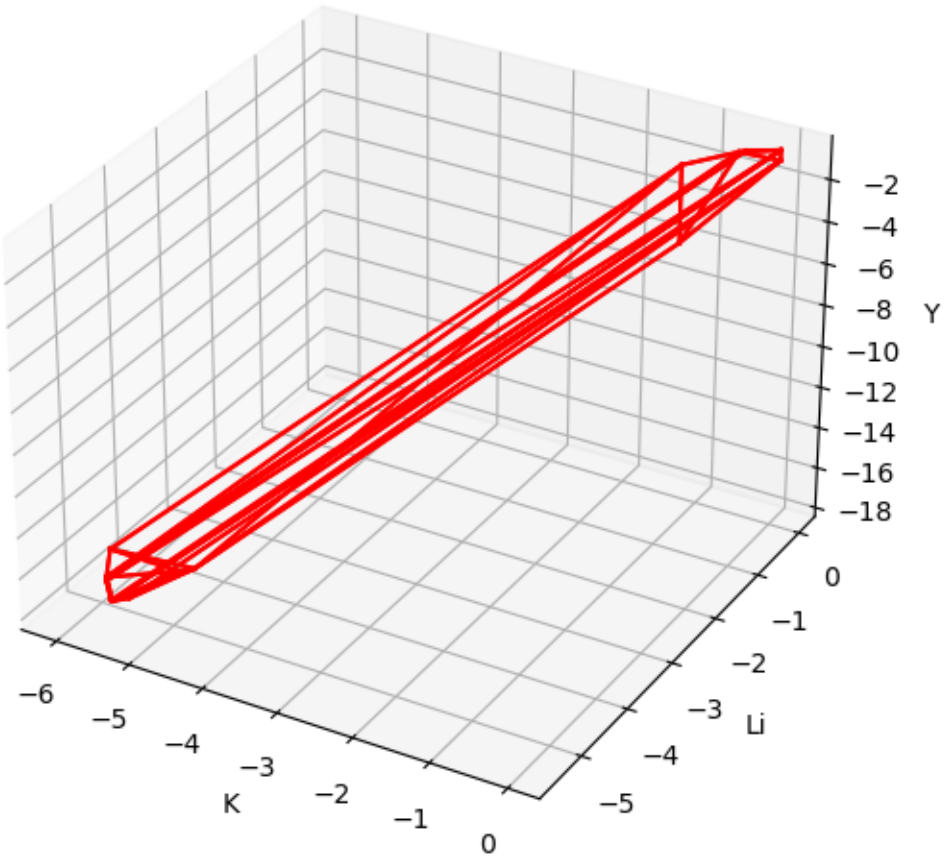


Fig: The 3D stable chemical potential region of K2LiYF6. (comes from the MP database)

Check out the files `stable_recalc_3d.out` and `fig-K2LiYF6_3d_recalc.png` . They are the image and

data output from the second calculation and analysis process.

K	Li	Y	F
-0.0059	0	-0.1727	-5.5633
-5.5692	-5.5633	-16.8626	0
-0.4569	0	-0.6237	-5.3378
-5.7947	-5.3378	-16.6371	0
-5.4204	-5.4145	-16.9064	-0.0671
-5.5546	-5.5487	-16.9064	0
-0.0059	0	-0.6629	-5.4816
-5.4204	-4.1335	-18.1874	-0.0671
-5.5546	-4.2677	-18.1874	5.92119e-16
-1.2869	0	-5.7869	-4.2006
-1.2869	1.56171e-17	-2.0624	-4.82135
-6.10825	-4.82135	-16.5265	0
-0.5489	0	-0.7709	-5.2826
-5.8315	-5.2826	-16.6187	5.92119e-16

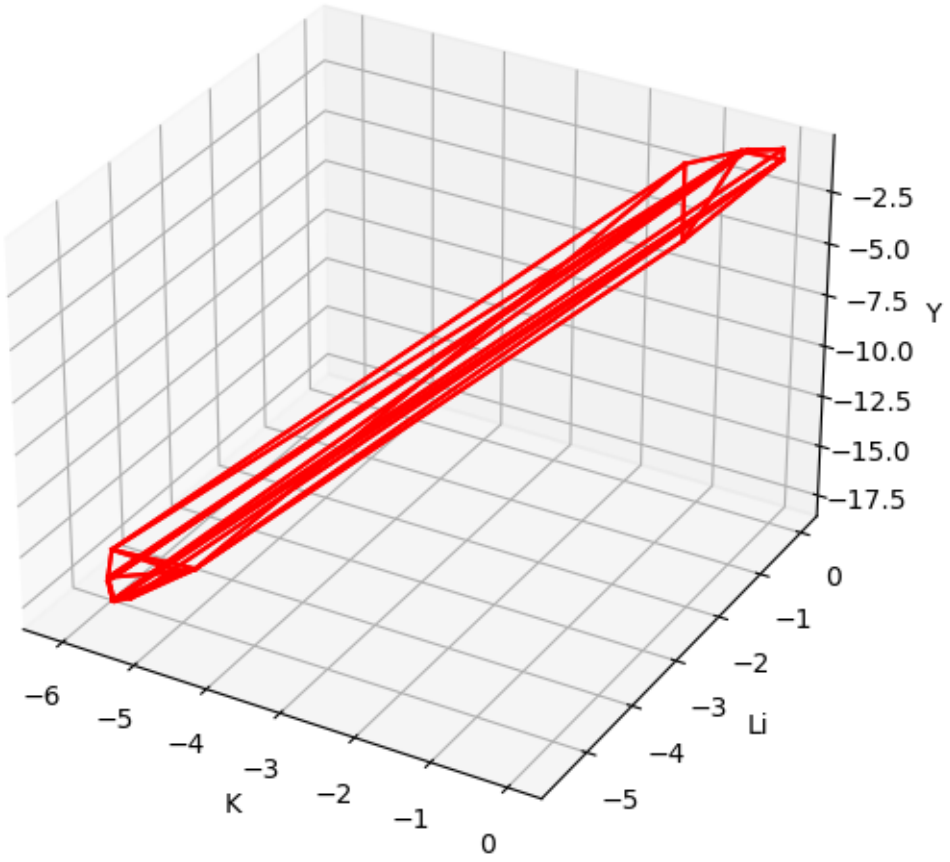


Fig: The 3D stable chemical potential region of K2LiYF6. (comes from the second stage calculations)

5.5.3 5.5.3 Rb2LiInI6 (Predicted result: unstable)

5.5.3.1 Prepare POSCAR and dasp.in

The `POSCAR` of Rb2LiInI6 can be obtained from MP database, and the structure needs to be optimized or set by the user. The `POSCAR` for this case is as follows:

```
Rb2 Li1 In1 I6
1.0
        7.7485766411         0.0000000000         0.0000000000
        3.8742883205         6.7104642143         0.0000000000
        3.8742883205         2.2368214048         6.3266863345
Rb   Li   In   I
2    1    1    6
Direct
0.750000000         0.750000000         0.750000000
0.250000000         0.250000000         0.250000000
0.500000000         0.500000000         0.500000000
0.000000000         0.000000000         0.000000000
0.750886977         0.249112993         0.249112993
0.249112993         0.249112993         0.750886977
0.249112993         0.750886977         0.750886977
0.249112993         0.750886977         0.249112993
0.750886977         0.249112993         0.750886977
0.750886977         0.750886977         0.249112993
```

Users need to set the relevant parameters in the file `dasp.in` according to their own situation, and set *tsc_only* = *T* and *database_api* . The details can refer to Cs2AgBiCl6.

5.5.3.2 Calculation and analysis

The total energy calculation of the host structure (the parameters are consistent with MP database):

TSC module will use the same input parameters (INCAR, KPOINTS, POTCAR) with the `Materials Project` database to perform structural relaxation and static calculation on the primitive cells given by the user. Therefore, the calculated total energy is comparable to that of the MP database. (refer to the previous cases)

Rapid analysis of the stability and key hetero-phases compounds:

The TSC module will search for all the secondary compounds that compete with Rb2LiInI6 in the MP database. According to the output file `materials_info.yaml` , it can be found that all the considered hetero-phases compounds including:

```
secondary_phases:
- - Rb
  - Li
  - In
  - I
  - InI4
  - InI3
  - InI
  - InI2
```

(continues on next page)

(continued from previous page)

```
  - LiInI4
  - Rb3InI6
  - RbInI3
  - RbInI4
  - LiI
  - RbI
  - RbI3
  - LiIn3
  - Li3In
  - LiIn
  - LiIn2
  - Li5In4
  - Li2In
  - Li13In3
  - Li3In2
  - Rb3In
  - RbIn4
  - Rb2In3
  - Rb8In11
  - Rb3Li
```

By comparing the calculated total energy of Rb2LiInI6 with that of the hetero-phases extracted from the database, it can be judged that Rb2LiInI6 is **unstable**. The relevant information can be seen in `2tsc.out` :

```
...
analysing the thermodynamic stability of Rb2LiInI6.
The stability of Rb2LiInI6 is: False.
Rb2LiInI6 may decomposed into ['RbInI4', 'LiI', 'RbI'].
you can set tag: 'excluded_phase' to get some reference values of chemical potentials.
analysing of Rb2LiInI6 is done.
...
```

The value of energy above hull of Rb2LiInI6 is output in file `materials_info.yaml`, it is positive and consistent with the result that "this compound is unstable" .

```
e_above_hull: 0.0819
```

The decomposed products include RbInI4, LiI, and RbI, and this also can be known from the output file `materials_info.yaml` :

```
decomp:
- RbInI4
- LiI
- RbI
```

As Rb2LiInI6 is unstable, TSC module can not judge the key hetero-phase compounds or output the images of the stable region. Moreover, TSC module will not do the calculation of the second stage further.

If users still want to do the second stage calculations to obtain the stable chemical potential region, they can set *excluded_phase* to exclude the effect of some hetero-phase compounds on the stability of the host compound based on the prompt in file `2tsc.out` . Generally, one or more of the decompositions is excluded to make the host compound has a referable stable region.

i.e. set *excluded_phase = RbInI4* , or *excluded_phase = LiI* , or *excluded_phase = RbInI4 RbI* in file

`dasp.in` , and run TSC module again. At this point, it is possible that the compound is still unstable, then some compounds in the current decomposition path can be appended to *excluded_phase* and re-run the TSC module. (refer to the analysis process of K2LiYF6.)

6

6 Common Problems and Solutions

6.1 6.1 DASP related

6.1.1 6.1.1 How to configure Pymatgen ?

- `Pymatgen` will be installed automatically when the user installs DASP. The pseudo-potential library in `Pymatgen` must be set to the 2003 version for VASP, otherwise, the results calculated by DASP program will be unreliable. However, the version of pseudopotential can be set arbitrarily in the DASP calculation. That is, `potcar_path` in `dasp.in` can correspond to any version of VASP pseudopotential.

6.1.2 6.1.2 How to place the sxdefectalign script

- Users can download `sxdefectalign.bz2` by themselves from the website: https://sxrepo.mpie.de/projects/sphinx-add-ons/files<https://sxrepo.mpie.de/projects/sphinx-add-ons/files, and make the program `sxdefectalign` executable, namely `chmod +x sxdefectalign` , and add its path to the environment variables.

6.1.3 6.1.3 What is the role of file `redo.in` in PREPARE and DEC calculations?

- Users can modify the parameters of `INCAR` when they encounter errors in VASP, and then write the calculation directory to the file `redo.in` under the dec directory, namely `dec/redo.in` , and run the program again.

6.1.4 6.1.4 Hard disk is full and program cannot execute right.

- The wavefunction and charge density will be output by default in the DASP calculation (users can manually modify `INCAR` after PREPARE calculation and choose not to write), therefore, large disk space is required. If the storage space is used up, the task system may crash.

6.1.5 6.1.5 Can *vasp_path_tsc* and *vasp_path_dec* be set to the same?

- These two parameters can be set to the same. But we suggest *vasp_path_dec* uses the `vasp_gam` version because DASP does not support multiple k points to do defect calculations currently (namely band filling correction is not included).

6.1.6 6.1.6 Can spin polarization and spin-orbit coupling be considered?

- `INCAR` generated by DASP by default includes spin-polarization calculation, namely *ISPIN=2* . If it is necessary to do the calculation including spin-orbit coupling, the path of VASP in `dasp.in` need to be changed to the `vasp_ncl` version, and modify the relevant parameters in `INCAR` after the PREPARE is completed.

6.2 6.2 Relevant issues in VASP calculation

6.2.1 6.2.1 VASP errors: issues related to *SYMPREC*

- Solution: Increase *SYMPREC* in `INCAR` and execute again. Users can manually set *SYMPREC* in `INCAR` after the PREPARE module is completed to avoid such errors.

6.2.2 6.2.2 The electronic step does not converge when using *level=2 or 3* to do HSE self-consistent calculation.

- Solution: Change the algorithm, such as the *ALGO* tag in `INCAR` could select *Damped or All* (do not mixed with Normal). Or increase *SIGMA* , this parameter should be reset after converged, and calculated again with reading the converged wavefunction.

6.2.3 6.2.3 The electronic step does not converge when doing relaxation1, relaxation2, or static calculation in the first stage of TSC.

- Solution: In the first stage of TSC, the parameters in `INCAR` used for DASP calculation (PBE) are completely consistent with MP database, so they may not converge in a few cases, in which only need to change the algorithm and set *ALGO=Fast or Normal* in `INCAR` .

6.2.4 6.2.4 What to do when the calculation stops but does not converge during structure relaxation in the DEC module?

- Solution: Generally, copy CONTCAR to POSCAR and rerun the calculation.