A VASP cookbook

Jinyuan Wu

January 14, 2022

This article briefly introduces VASP, its features, its frequently used functions, frequently observed errors and unexpected results, and some tricks to speed up calculation.

1 Ab-initio calculation using VASP

VASP [1,2], or Vienna Ab initio Simulation Package, is a popular commercial *ab initio* package written in Fortran, supporting plain-wave Kohn-Sham DFT and related approaches for electronic structure and molecular structure investigation. The most frequently used function of VASP is plain-wave based Kohn-Sham DFT, with three kinds of pseudopotentials supported, namely norm-conserving pseudopotentials, ultrasoft pseudopotentials and PAW pseudopotentials. VASP also support other self consistent algorithms that are not DFT, for example GW or RPA calculation, but usually a DFT calculation must be performed ahead to generate necessary auxiliary data like wavefunctions. By employing forces on atoms calculated from DFT outputs, VASP supports crystal structural relaxation as well as *ab initio* molecular dynamics.

The best way to learn VASP is to read its own manual and examples, which are available on [5]. It is, nonetheless, also the worst way to lean VASP, because a beginner will be crushed by the flood of configuration tags and strange terms. This article serves as an introduction as well as a quick reference for VASP, in which we present an integrated discussion concerning the underlying theories, input and output files, common tasks, frequent mistakes and tricks.

1.1 Plain-wave KSDFT

We are not going to introduce the formalism of **plain-wave based Kohn-Sham DFT** or KSDFT (which may be found in the solid physics notes, Section 7.1), and just present what it is: in a word, in a crystal system, KSDFT means solving the following equations: the eigenvalue problem **Kohn-Sham equation**

$$-\frac{\nabla^2}{2m}\phi_{n\boldsymbol{k}} + \underbrace{\left(\int \mathrm{d}^3\boldsymbol{r}'\,\frac{\rho(\boldsymbol{r}')}{|\boldsymbol{r}-\boldsymbol{r}'|} + \int \mathrm{d}^3\boldsymbol{r}'\,V_{\mathrm{PP}}(\boldsymbol{r}',\boldsymbol{r}) + V_{\mathrm{XC}}(\boldsymbol{r})\right)}_{V_{\mathrm{eff}}}\phi_{n\boldsymbol{k}}(\boldsymbol{r}) = \epsilon_{n\boldsymbol{k}}\phi_{n\boldsymbol{k}} \qquad (1)$$

and

$$\rho(\boldsymbol{r}) = 2\sum_{n,\boldsymbol{k}} f_{n\boldsymbol{k}}|\phi_{n\boldsymbol{k}}|^2, \qquad (2)$$

where $\boldsymbol{k}$ is the Bloch wave vector, n labels different solutions with the same $\boldsymbol{k}$,

$$V_{\mathrm{XC}}(\boldsymbol{r}) = 2\frac{\delta E_{\mathrm{XC}}[\rho(\boldsymbol{r})]}{\delta\rho(\boldsymbol{r})}, \qquad (3)$$

$E_{\mathrm{XC}}[\rho(\boldsymbol{r})]$ is a functional of electron density named **exchange-correlation functional**, V_{PP} is the **pseudopotential** which is a smoothened version of atomic potentials in the crystal, and $f_{n\boldsymbol{k}}$ is a given function of $\epsilon_{n\boldsymbol{k}}$s, which may be Fermi distribution or Gaussian smearing. Note that in the original Kohn-Sham formalism, $f_{n\boldsymbol{k}}$ is a step function, but usually we will introduce *smearing* to make it smooth for better convergence. Although VASP works at $T = 0$, introduction of fractional occupation causes an *imaginary* temperature and the corresponding TS term (a natural result for additional constraints on the optimization problem), and when smearing is introduced, what is minimized is *not* the total energy but the free energy, so *always double check which "energy" are you using*. It should also be noticed that the "free energy" here has nothing to do with the free energy in a real thermal system, say, in a molecular dynamics simulation.

Above is the version of spinless KSDFT. We also have version of KSDFT in which electrons can have parallel spins or non-parallel spins, or relativistic effects like spin-orbital coupling play a role, all with similar formalisms.

When the calculation is done, $\rho(\boldsymbol{r})$ (or spin density matrix when spins are involved) gives the correct electron density, and the **energy functional**

$$E[\rho(\boldsymbol{r})] = \sum_{n,\boldsymbol{k}} f_{n\boldsymbol{k}} \int \mathrm{d}^3\boldsymbol{r}\, \phi_{n\boldsymbol{k}}^*(\boldsymbol{r}) \left(-\frac{\nabla^2}{2m} + \int \mathrm{d}^3\boldsymbol{r}'\, V_{\mathrm{eff}}(\boldsymbol{r},\boldsymbol{r}') \right) \phi_{n\boldsymbol{k}}(\boldsymbol{r}) \tag{4}$$

gives the ground state energy, which can also be obtained by

$$E = \sum_{\text{occupied } n,\,\boldsymbol{k}} \epsilon_{n\boldsymbol{k}}, \tag{5}$$

or by

$$E = F + TS, \tag{6}$$

where F is the free energy, and T measures the strength of smearing. For a Fermi-liquid system the functions $\phi_{n\boldsymbol{k}}(\boldsymbol{r})$ (named **Kohn-Sham orbitals**) and the eigenvalues $\epsilon_{n\boldsymbol{k}}$ are *roughly* single electron wave functions and energies, but their exact meaning is not clear and sometimes may not be a good approximation. For tasks requiring high accuracy of electronic structures it is recommended to use GW or RPA methods with Kohn-Sham orbitals and eigenvalues as the input.

The Kohn-Sham orbitals are represented using a group of plain-waves as a basis. The free particle kinetic energy corresponding to the highest $\boldsymbol{G}$ is called the **cutoff energy**

$$E_{\mathrm{cutoff}} = \frac{\boldsymbol{G}^2}{2m}. \tag{7}$$

The higher the cutoff energy is, the finer spacial details of Kohn-Sham orbitals can be.

To quicken the solving process, VASP uses an iterative diagonalization scheme, the details of which can be found in [3].

1.2   What VASP doesn't do

Here is a list of things not yet supported - and highly unlikely to be in the future - by VASP:

- Thermal DFT;

- KSDFT with Gaussian basis or other non-plain-wave basis;

- Orbital-free DFT;

- Density matrix-based DFT, or DFT for symmetry broken systems or strongly correlated systems;

- Post Hartree-Fock methods.

The role

2   Input files

Four files *must* be in the directory where `vasp` command is run. They are `INCAR`, `POSCAR`, `POTCAR` and `KPOINTS`. `INCAR` controls the simulation: it specifies parameters in the calculation, including the cutoff energy, the

2.1   INCAR

All tags in `INCAR` can be found in [4].

2.1.1 Tags that work well with default values

Usually you *don't* need to specify what functional is used because it can be read from `POTCAR`. Every `POTCAR` contains the name of the functional with which it is generated, and since a pseudopotential generated with one functional should not be used with another functional, VASP uses the functional recommended by the `POTCAR` file.

2.1.2 Tags that must be treated with care

Some tags *need* to be set with caution or otherwise problem will keep popping up.

`LMAXMIX` must be set according to the orbital with highest l when doing DFT+U.

2.1.3 Confusing tag names in `INCAR`

Due to some historical reasons, some tags in `INCAR` have misguiding names. To name a few:

- The `GGA` tag can be used to apply an *LDA* functional. For example you can set `GGA = CA` or `GGA = PZ`.

- The `LDAU` tags - `LDAUU`, `LDAUJ` and `LDAUL` - *don't* require an LDA functional. They actually control *GGA+U*.

2.2 `POSCAR`

2.3 `POTCAR`

For example, to create a `POTCAR` for $BaFe_2As_2$, use the following command:

```
cd the/place/where/you/store/pseudopotentials
cat POTCAR_Ba POTCAR_Fe POTCAR_As > POTCAR
mv POTCAR the/place/where/you/run/VASP
```

Make sure all pseudopotentials you use to simulate one material are of the same type and the same functional.

2.4 Frequent mistakes in input files

Fatal mistakes in input files usually kill a VASP process in several seconds, and for some reason, the error message is sometimes hard to understand, and sometimes it is even hard to notice that the calculation *already ends.* Whenever something strange happens, before anything else, do the following checklist:

- Are these really the input files used in your last job? People often make stupid mistakes such as modifying some settings but using the unmodified version of input files to submit another job, so the errors in the last run remain there, leaving themselves confused about why the new settings don't work. Always check:

  - The `SYSTEM` tag in `INCAR`. It's quite often when people copy the `INCAR` for one material to calculate another system, and forget to change `SYSTEM`. So you are assuming that you have edited the `INCAR` for one material while you are actually editing another one.
  - The last modified date of every file. If you get unexpected results it may be the remaining of the last calculation, and this round of calculation ends because of a mistake in input files.
  - Whether there is some technical mistakes in, for example, SLURM scripts, so the jobs with the new settings never run.

- Make sure all the four necessary files are in the working directory. Unconsciously deleting important files is not rare.

- Make sure that atoms are in the correct order. Make sure the orders of atomic species in `POTCAR` and `POSCAR` are the same.

Most of the items in the list are apparently stupid mistakes, but people do make such mistakes from time to time. So whenever something goes wrong, *always* check *everything* in this list.

3 Output files

3.1 `OUTCAR`

The static configuration has the point symmetry

3.2 `vasp.out`

Note that the `E` column in `vasp.out` is *not* the expectation of energy. Instead, it is the *free energy*.

An example of electronic steps are listed here:

```
       N        E                     dE             d eps         ncg       rms             rms(c)
DAV:   1    -0.156969645813E+02   -0.15697E+02   -0.27531E+00 33936   0.335E+00
DAV:   2    -0.156982451091E+02   -0.12805E-02   -0.12805E-02 52360   0.317E-01
DAV:   3    -0.156982452245E+02   -0.11543E-06   -0.11559E-06 37800   0.167E-03
    1 F= -.15698245E+02 E0= -.15693289E+02  d E =-.991302E-02
```

You may notice that

3.3 Criteria of whether something goes wrong

If everything goes smoothly, usually the calculation converges in 40 or 50 electronic steps. It is, therefore, a good idea to keep `NELM` in its default value or a slightly larger value.

4 Frequent tasks

4.1 Calculating the ground state energy

4.2 Calculating the band structure

Static self-consistent calculation

4.3 Structural relaxation

Suggested `INCAR` setting:

- `EDIFF = 1e-8`, `EDIFFG = -0.0001` for better accuracy.

- `ISMEAR =`

4.4 Phonon spectrum

Phonon spectrums can be calculated in a number of ways:

- **Frozen phonon method**,

4.5 Molecular dynamics

5 Fixing errors

5.1 Errors during calculation and how to fix them

Here is a list of errors that may be encountered during a calculation with a mostly proper `INCAR`:

- `ZBRENT: fatal error in bracketing`: copy `CONTCAR` to `POSCAR` and try again with smaller `EDIFF`, and, if it does not work, use `ADDGRID`.

It seems `ISYM = 3` is less tolerant than `ISYM = 2`.

5.2 Unexpected outputs

Whenever something weird happens, check these terms:

- Are there too few or too many k points? Check `IBZKPT`. If there are, say, only 2 k points, you won't expect the result to make sense.

- Does the calculation really converges? Check `vasp.out`.

- Maybe the optimization algorithm doesn't work well for the given task. `ALGO = Normal` is always a safe choice, though it may not be that efficient. But when other algorithms actually fail, `ALGO = Normal` at least will finish with a reasonable time duration and give an acceptable result.

- If `POSCAR` is modified manually (for example, to symmetrize the structure), always do a double check to ensure you didn't accidentally introduce a wrong structure.

- There are tags which assign one value for each atom. If the number of assigned values are fewer than the number of atoms in POSCAR, VASP will terminate and report an error, but if the number of assigned values are *more* than needed, often there is *no* error but the calculation is completely wrong. Such tags include:

    - `MAGMOM`

5.2.1 Strange `vasp.out` outputs

Why are there so many lines beginning with "soft electron" or "augmentation electron"? With `LAECGH` turned on, you will find lines starting with `soft electron` or `augmentation electron` in `vasp.out`.

5.2.2 Charges and magnetizations in `OUTCAR`

The total charges and magnetization doesn't match the numbers in `vasp.out`! These 'total charges' and magnetizations are obtained form a summation over the local quantities (integration of the charge and spin densities in the 'atomic spheres' of radii r=`RWIGS`. As the volumes of the spheres do not sum up to the volume of the unit cell, these charges also do not sum up to the total number of valence electrons (the contribution of the "interstitial" space is missing).

5.2.3 Energy

5.3 Unreasonable results

Systems containing Fe are usually hard to deal with.

6 Why the calculation is slow and how to speed up

Note that soft pseudopotentials are just designed for small `ENCUT`. Increasing `ENCUT` does not increase accuracy for them. Hard pseudopotentials are required for localized electrons and/or anything involving Hartree-Fock approximation, for example hybrid functions.

6.1 Electronic convergence

6.1.1 Lattice structure

A insufficiently relaxed lattice structure usually comes with poor convergence performance.

6.1.2 Magnetic orders

It should be noted that though `ISPIN = 2` seems to be more expansive than `ISPIN = 1`, for a system in which spins play an important role in relaxation, using `ISPIN = 1` *won't* speed up calculation. So if you are struggling with `ISPIN = 1` on a specific system, maybe the system has a magnetic order or something so that its lattice structure is designed to induce an electronic order with magnetization. In this case, `ISPIN = 1` usually takes longer time to converge.

7 Auxiliary tools

7.1 Phonopy

Phonopy can also be used to symmetrize a certain supercell. First use a command like `phonopy -c POSCAR --symmetry --tolerance=1e-3` to obtain the (symmetrized) unit cell, and then find the correct `DIM` matrix that expand the unit cell into the expected supercell.

8 Scaling up

Golden rule: *always* keep record of a file that you will probably use in the future. Take down under which settings is it obtained. Take down the date. Take down remarkable obstacles you face trying to get it.

References

[1] G. Kresse and J. Furthmüller. Efficient iterative schemes forab initiototal-energy calculations using a plane-wave basis set. *Physical Review B*, 54(16):11169–11186, October 1996.

[2] G. Kresse and J. Furthmüller. Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set. *Computational Materials Science*, 6(1):15–50, 1996.

[3] The VASP team. Algorithms used in vasp to calculate the electronic groundstate. https://www.vasp.at/wiki/index.php/Algorithms_used_in_VASP_to_calculate_the_electronic_groundstate.

[4] The VASP team. Category:incar. https://www.vasp.at/wiki/index.php/Category:INCAR.

[5] The VASP team. The VASP Manual. https://www.vasp.at/wiki/index.php/The_VASP_Manual.