

VASP

I include here some accumulated VASP wisdom. Many of these tips are rules-of-thumb, so consider investigating them for your particular system of interest.

Geometry Optimizations

- Don't waste your time using super high-accuracy settings on a structure far from the local minimum. Do an initial optimization with "looser" settings (e.g. gamma-point only for the k-point grid, 400 eV cutoff if relaxing the atomic positions) and then refine it with the desired parameters.
- When performing geometry optimizations that involve changes in the cell shape and/or volume, always ensure that $ENCUT \geq 1.3 \cdot ENMAX$ to prevent **Pulay stresses**. For structures containing carbon, this often means using a minimum cutoff of 520 eV.
- The first step in a geometry optimization will generally have the highest number of SCF iterations. It is okay if that first step does not converge electronically within the limits of NELM. In fact, it is better to have the first step reach NELM instead of running for many hundreds of SCF iterations.
- The general rule-of-thumb is to set EDIFFG to 0.05 eV/Å or lower for optimizations. I personally recommend 0.03 eV/Å or lower. For large, flexible materials, any value greater than 0.03 eV/Å is likely too high. Remember that in VASP, the EDIFFG flag must be a negative value if you want a convergence threshold based on the maximum net force, not the change in total energy.
- When performing a full optimization of atomic positions and cell shape/volume, it is almost always best to do this in stages. It is often wise to start with a relaxation of the atomic positions (ISIF=2) followed by a full volume relaxation (ISIF=3). This will significantly reduce the chance of running into convergence issues.
- If your material can only be modeled with vacuum space as part of the simulation unit cell (e.g. a surface slab), never use ISIF=3 to optimize the cell volume, as it will simply reduce/eliminate the vacuum. You need to vary the lattice parameters manually and find the global energy minimum that way.
- When performing optimizations, you are probably better off setting ISYM=0 unless you are certain you wish to constrain the symmetry. Even though this will increase the computational cost by not using available symmetry, it allows the system to reach local minima that would not otherwise be accessible. In addition, it prevents you from having to worry about errors related to SYMPREC. That being said, some symmetry issues sometimes still arise despite having ISYM=0. To resolve these, set SYMPREC=1.0e-8.
- If you wish to stop a job but want it to still output any WAVECAR or CHGCAR files, create a file in the working directory named STOPCAR and have a line that reads LSTOP=.TRUE. (this will cause the job to stop on the next ionic step, and all restart files will be written)

Choice of Optimization Algorithms

- In choosing an optimization algorithm for finding local minima, I generally recommend starting with the conjugate gradient (CG) algorithm (IBRION=2) because it is very robust and you do not have to worry about tweaking POTIM.
- In large, flexible materials with many degrees of freedom, the CG optimization algorithm (IBRION=2) oftentimes results in a bracketing error once it gets relatively close to the local minimum (search for ZBRENT: fatal error in bracketing in the standard output file). This occurs because the potential energy surface is very flat, and the CG algorithm implemented in VASP is based on the energy differences. One option to fix this is to copy the CONTCAR to the POSCAR and tighten EDIFF to 1e-6, but a more reliable option is to use a force-based optimizer. Of these, I'd recommend FIRE as implemented with VTST (IBRION=3, IOPT=7). I have found that FIRE is generally more robust than the QN (IBRION=1) method.
- If the starting structure has extremely high forces on some atoms, make sure you use an appropriate optimizer, at least for the first few steps. I strongly recommend a force-based optimization algorithm with a robust line-search algorithm. For cases where the forces are so high that the structure "explodes" within a few iterations, I recommend using ASE's BFGSLineSearch algorithm until $\max |F| < 10$ eV/Å or so. Then continue with your favorite optimizer.
- Generally, for NEB and CI-NEB calculations, the L-BFGS algorithm (IOPT=1) implemented in VTST is the fastest. For the dimer method, the force-based CG method in VTST (IOPT=2) is recommended. However, if you are having trouble in either case, I suggest switching to the FIRE algorithm (IOPT=7) with the default settings. It is a bit slower, but it is especially useful in troublesome cases of convergence.

Electronic Energy Convergence

- For insulating materials, SCF convergence is greatly accelerated by using `ALGO=All`. This has the added benefit that you don't have to worry about any of the mixing tags. This algorithm is also recommended when using the M06-L meta-GGA functional, regardless of system type.
- I recommend setting `LWAVE=.TRUE.` for most jobs, unless you are using an ASE-based optimizer or running many short jobs in succession. There is a large file I/O cost, but the `WAVECAR` serves as an excellent starting guess for continuation jobs, if needed, and can be deleted when you're done. I don't necessarily recommend setting `LCHARG=.TRUE.` for optimizations. The initial charge density can be calculated from the previous `WAVECAR`.
- Be careful about obtaining final energies from calculations using real-space projectors (e.g. `LREAL=Auto`) even for large systems. This can impact energies somewhat (even energy differences) depending on what you wish to study. If VASP recommends in the `OUTCAR` file that your system is large and that you will save time by using real-space projectors, I strongly recommend running your calculations with `LREAL=Auto` and then doing a final optimization with `LREAL=False`. The optimization with the reciprocal-space projectors will only require an additional 1-5 steps on average, so you will still get the speedup of using real-space projectors.
- While the VASP manual suggests not setting the maximum number of SCF iterations (`NELM`) above 60, there are many materials (e.g. MOFs) where a higher value is needed for the first few steps. I generally set `NELM=150` when studying very large materials.
- For materials that have charge sloshing or difficult convergence issues with the SCF, I recommend setting `NELMIN` to 4-6 so that accurate energies and forces are obtained.
- If you do not explicitly set `ISTART` or `ICHARG`, VASP will detect if a `WAVECAR` is present and use this for the initial wavefunction and charge density if it exists. This is arguably better than explicitly setting `ISTART` in your `INCAR` because it prevents VASP from crashing if a `WAVECAR` is not present (e.g. because the previous job crashed).
- For insulating materials (or materials that have unknown band gaps), I recommend using Gaussian smearing (`ISMEAR=0`) since it is appropriate for both conductors and insulators, although there are better choices for conducting materials.
- For insulating materials, I recommend settings `SIGMA` to 0.01 eV as a reasonable initial test. Always check to make sure that extrapolation back to $T = 0$ K from the finite-temperature approximation is close to the fictitious free energy, as listed in the `OUTCAR` file.
- It is generally good practice to set `PREC=Accurate`. This provides reasonable estimates for the integration grid (`NGX`, `NGY`, `NGZ`). It is often stated the meta-GGA functionals require tight integrations grids to achieve appreciable convergence. I have found that `PREC=Accurate` is sufficient even for these troublesome functionals.

Spin Polarization

- When studying systems with magnetic moments, set `LORBIT=11` so that you can view the converged magnetic moments for each atom.
- If you are continuing a job from a previous `WAVECAR`, you do not need to set the `MAGMOMS` in the `INCAR` file so long as `ISPIN=2` (you can if you'd like, but VASP will ignore them).
- If you are unsure what to set as the magnetic moment for a metal atom, consider setting the `MAGMOM` value to the number of anticipated unpaired electrons. If this too is completely unknown, setting a value of 5 for *d*-block and 7 for *f*-block elements is typically okay as a first guess. If this approach is taken, it would be wise to use the converged structure to also test other spin initializations. I would also test an initial `MAGMOM` of 0.1 to see if this converges to a system with no spin, back to the same magnetic moment as the high-spin initialization, or something else entirely.

Transition States

- If upon restarting a dimer calculation you find that the torque and angles are higher than where they left off, make sure that you are using a VASP build with VTST 3.2 or newer. See [here](#).
- The usage instructions for the VTST `dimmins.pl` script are incorrect. It should be `dimmins.pl POSCAR MODECAR displacement`, where the `POSCAR` and `MODECAR` are the resulting files from doing `vfin.pl`
- The usage instructions for the VTST `neb2dim.pl` script are unclear. It should state that you must first run `vfin.pl`, copy the `exts.dat` file to the parent directory (where the new `POSCAR` files are written), and run `neb2dim.pl` from that parent directory.

Spin/Charge Densities

- Never generate `AECCAR` files (e.g. for a Bader analysis or DDEC charges) during a geometry optimization. VASP writes the `AECCAR0` file for the input geometry but

the AECCAR2 file for the converged geometry, so they are not compatible. Always do a separate single-point energy calculation (nsw=0) when setting LAECHG=.True.

Pseudopotentials

- Always consider using the VASP-recommended PAW potentials, shown [here](#). Note that Li_sv has an ENMAX of 499 eV in the 5.4 version of the PAW_PBE pseudopotentials, but all the rest should have $\text{ENMAX} \leq 400$ eV. There is now an option in ASE to automatically choose the VASP-recommended default pseudopotentials. Just set `setups='recommended'`.
- For DFT+U, hybrid functionals, and meta-GGA calculations, it is [recommended](#) to set `LASPH=.TRUE.` If you use the `LASPH=.TRUE.` flag, make sure you include it for gas-phase species as well! It should be considered as part of your model chemistry.
- A subtle point is that, because the non-spherical contributions to the gradient corrections inside the PAW spheres will change with ENCUT, it will be hard (if not impossible) to converge an absolute energy with respect to the plane-wave kinetic energy cutoff if `LASPH=.TRUE.` Instead, one should look at convergence of relative energies or do a convergence test with `LASPH=.FALSE.`

Parallel Performance

- The rule-of-thumb is that the number of processors you should use scales reasonably well with the number of atoms in the system. This holds fairly well for metallic systems where there are many electron-rich elements, but can be a slight over-estimate for organic or organometallic systems where there are many C and H atoms.
- Whenever possible, use the gamma-point only VASP executable, as it runs up to 1.5 times faster than the standard executable.
- I recommend using NCORE in place of NPAR since it automatically adjusts based on the number of nodes. The optimal value for NCORE strongly varies based on the computing environment. A good first-guess on many compute clusters is to try setting NCORE to the number of processors in a given node. Regardless, always make sure that the number of processors per node is divisible by NCORE.

ASE

- Whenever possible, refrain from using the ASE optimizers (or any external optimizer) with VASP calculations. If necessary, consider using the `VaspInteractive` calculator instead of the `Vasp` calculator, as this will reduce CPU time associated with starting and stopping VASP every ionic step. However, even if the `VaspInteractive` calculator is used in place of the `Vasp` calculator, VASP will generally require more electronic steps when run with an ASE optimizer than with an internal VASP or VTST optimizer. This is because VASP keeps an internal history of previous ionic steps in its mixer history and because VASP extrapolates the wavefunction and charge density from the previous step to the new positions. There is also the I/O overhead associated with reading/writing any restart files every ionic step.
- The above holds true for vibrational frequency calculations as well. That being said, it sometimes may be worth it to use ASE's `vibrations` module even if there is added CPU overhead. The ASE-generated vibrations can be easily visualized using the ASE GUI, whereas VASP-generated vibrations are more difficult to visualize. The ASE `vibrations` module also saves restart files (`.pck1` files) for every displacement, so if the job crashes or exceeds the specified walltime, it is easy to continue where you left off.