

EXERCISE 8

ORDER-N SOLVER

In this exercise, you will use the Order-N solver in Siesta. The system consists of a cluster of 4 H₂O molecules, which have been relaxed previously. The input file *h2o_4.fdf* is setup to do a single point (static) calculation, but you can also run some dynamics if you feel like it.

The input file uses a basis set of quite good quality (DZP, with relatively long orbitals), which is necessary to describe properly the internal structure of the H₂O molecule and the weak H-bonds.

In this exercise you will use the Order-N solver, using the functional of Kim-Mauri-Galli.

1. Convergence as a function of Localization Radius

Run the calculation for different values of the localization of the Wannier Functions, from 1.5 Ang to 5.0 Ang. Look at the outputs, and plot the total energy as a function of localization range. Do you see the exponential behaviour? Calculate the total energy of a single H₂O molecule (using input file *h2o.fdf*) and plot the value of the binding energy of the cluster as a function of localization range. Determine what is the minimum value of the localization radius that you must use if you want an accuracy of 1% in the binding energy of the cluster.

2. The 'Chemical Potential': possible instabilities

In the Kim-Mauri-Galli functional, it is necessary to know the value of the Chemical Potential or Fermi level (*ON.eta* parameter), which determines which states are occupied and which ones are empty. This must be a value within the band gap of the system. If *ON.eta* is not in the band gap, the computed number of electrons will not be correct, and, even worse, the algorithm may become unstable, and the run will crash.

The input file *h2o_4.fdf* you have used so far, the Chemical Potential is estimated by SIESTA, in O(N) operations, using the projection method of Goedecker. See all the flags called *ON.ChemicalPotential** in the fdf file, and check their meaning in the User's Guide. You have observed that the runs are stable, because the value of *ON.eta* is correctly computed by the code.

However, computing the Chemical Potential can be relatively time-consuming, so in many cases (specially for long runs in large systems) you may want to avoid it, if possible. If you have a good estimate for *ON.eta* (from a previous calculation, or from a previous MD step), you can avoid computing it, saving CPU time. To do that, you just need to put the *ON.ChemicalPotentialUse* and the *ON.ChemicalPotential* variables to *.false.* and introduce the value of the Chemical Potential using the *ON.eta* variable.

A typical case when you need to use the automatic estimation of the Chemical Potential is at the very beginning of a calculation, when you do not have an input *DM* or *LWF* file. In that case, usually you do not have an idea of where the Fermi level is. But, even if you do know it, the position and width of the gap may change abruptly during the first steps of the SCF cycle. In this case, it is advisable to use the *ON.ChemicalPotentialUse* option.

To see this kind of situation, and to show you a case where instabilities due to a wrong position of the Chemical Potential can arise, you can try to run the *h2o_4.fdf* system, but removing the *ON.ChemicalPotential* options, and defining explicitly the value of *ON.eta* in the input file. Can you find a value of the *ON.eta* for which the calculation is stable?? Pay attention to the number of electrons after each SCF cycle.

Now, you can try to start a calculation reading a *DM* from a previous, converged calculation in which you used diagonalization. See what is the value of the Fermi level from the diagonalization run, and use it for the *ON.eta* parameter in the Order-N run. Remember to read the *DM* setting the *DM.UseSaveDM* to .true. What do you find? Is the run now stable?