Convergence tests

For numerical computations it is imperative to check certain algorithmic parameters for "convergence". In general, this implies that the same system is calculated repeatedly while applying different values for these parameters until a certain output property does not change anymore. In that case, a parameter is called "converged".

In standard DFT electronic structure calculations (i. e. plane wave basis set and Monkhorst-Pack Brillouin zone sampling), the two main parameters which are imperative "to converge" are the plane wave energy cut-off (`ecut`) and the grid of wave vectors in reciprocal space (`kpts`). The property which should not change after a suitable set of parameters has been found is usually chosen to be the total energy (or in GPAW language: `potential_energy`).

1. Check convergence for your system with respect to the aforementioned parameters. First define a set of reasonable values, e. g. for the energy cut-off a range between 200 and 1200 eV and a uniform grid of $n \times n \times n$ points with n in the range from 4 to 16.
2. Use a for-in-loop to iterate through the energy range in a pythonic way. What energy cut-off is at least necessary to "converge" the system to a total energy difference of less than 2 meV per atom?
3. Use the same script again and adapt it for testing convergence with respect to the k-grid.
4. After analyzing the output numerically, what can you tell about:
   - Values of converged parameters for your system compared to the other systems
   - Time consumption when increasing the reciprocal grid size or energy cut-off
5. Visualize the data using matplotlib