

Exercise Ex7

December 1, 2025

Student: Romain Couyoumtzélis Sciper: 340933

Please use this template to submit your answers.

If you had to modify code from the notebook, please include the modified code in your submission either as screenshot or in a

```
\begin{lstlisting}[language=Python]
\end{lstlisting}
```

environment.

You only need to include the code cells that you modified.

Note, that references to other parts of the documents aren't resolved in this template and will show as `??`. Check the text of the exercises on website for the reference

Exercise 1

Fix the error in the single-point wavefunction optimization for the allyl radical using the BLYP exchange-correlation functional.

Here we changed 1 2 to 0 2 in the definition of the molecule :

```
1 symmetry c1
2 0 2
```

Indeed the original code is invalid for a neutral radical. The charge must be 0 and the multiplicity is 2.

Exercise 2

Fix the error in the single-point wavefunction optimisation of the $Cu[H_2O]^{2+}$ aqua complex (as observed in the gas phase) using the PBE exchange-correlation functional.

The calculation was set up with an open-shell Kohn–Sham reference (`reference = 'uks'`), which is appropriate for the doublet $Cu[H_2O]^{2+}$ complex, but the SCF failed to converge within the default 100 iterations when using the very small 3-21G

basis set. This basis is not designed for transition metal cations and provides an insufficiently flexible description of the Cu 3d shell and the lone pairs on the water.

In order to get more flexibility, we increased the basis set size to 6-31G and add polarization functions on heavy atoms such that we allow orbitals to stretch in the direction of $Cu - O, O - H$ bonds and better represent lone pairs.

We changed :

```
1 psi4.energy('pbe/3-21G', molecule=cu_h2o_2plus)
for :
1 psi4.energy('pbe/6-31G*', molecule=cu_h2o_2plus)
```

Exercise 3

Fix the error in the geometry optimisation of N_2 using the BP86 exchange-correlation functional.

Since we want to perform geometry optimization, we need to perform optimization before computing the energy, we therefore changed :

```
1 psi4.core.clean()
2 psi4.core.set_output_file('n2_opt.log', False)
3 psi4.energy('bp86/6-31+G*', molecule=n2)
for :
1 psi4.core.clean()
2 psi4.core.set_output_file('n2_opt.log', False)
3 psi4.optimize('bp86/6-31+G*', molecule=n2)
4 psi4.energy('bp86/6-31+G*', molecule=n2)
```

Exercise 4

Fix the error in the single point calculation of 3O using the B97-2 exchange-correlation functional.

Since 3O is an open shell system because two electrons are unpaired, we can't describe that with a restricted method. We need a spin-polarized description where the α and β densities are different.

We changed :

```
1 psi4.core.set_output_file('o3_singlept.log', False)
2 psi4.energy('b97-2/6-31+G*', molecule=o)
for :
1 psi4.core.set_output_file('o3_singlept.log', False)
2 psi4.set_options({'reference': 'uks'})
3 psi4.energy('b97-2/6-31+G*', molecule=o)
```

Exercise 5

Before running the calculations, give your best estimate as to which of the reaction energies (i.e. the energy difference between product and reactants) do you think is most "difficult" to calculate? Rank them in order of

increasing difficulty and explain the reasoning behind your choices.

Difficult in this case refers to getting an accurate reaction energy with low absolute error to true energy (reference energy from the dataset). As chemical accuracy we usually use 1 kcal mol⁻¹

For this question, I would rank the reactions in the following way based on the diffulty for each to achieve ~ 1 kcal mol⁻¹ of accuracy on the reaction energy:

1. **Methanol: staggered / eclipsed** (easiest).

It is a relatively small and closed-shell organic conformers; the energy difference is governed by relatively local effects. Such conformational profiles are well described by most modern DFAs and are extensively represented in training and benchmark sets.

2. **CaO bond dissociation to Ca + O** (intermediate).

CaO has an homolytic bond dissociation that only involves breaking a single covalent bond for diatomic molecule. This corresponds to a single-reference system for which DFT can easily describe its open-shell fragments with reasonable accuracy. The main difficulty is treating the atomic spins correctly once the bond is broken, while dispersion effects remain negligible.

3. **Ethane dimer interaction energy** (hardest).

For the ethane dimer, the interaction is the hardest to model reliably because it is governed by long-range dispersion forces. Conventional GGA and hybrid functionals that lack explicit dispersion corrections systematically underestimate this binding.

Exercise 6

Run the calculations below and check whether your assessment of the hard, medium and easy reactions is correct. Include the tables containing the calculated energies for the three problems in your report. Which functional do you think contained the reaction in its training data?

If we look first the Methanol case, it is worth noting that all methods converge within the chemical accuracy of 1 kcal/mol with absolute error always ≤ 0.25 kcal/mol. This confirms that this method is the easiest one. Secondly, for the the study of Ethane Dimer and Monomer, we are in the intermediate case where some methods are within the chemical accuracy. Methods without an explicit dispersion treatment (B3LYP, PBE, SVWN) perform poorly, whereas B3LYP-D3(BJ), B3LYP-NL, PBE-D3 and MP2 yield interaction energies close to the reference and within chemical accuracy. The best results for this case, is given by b3lyp-d3(bj). Indeed it allows to incorporate dispersion through empirical potentials (2 and 3-body), while the BJ method makes the potentials finite. To represent the behaviour of the interaction between the monomers,

	E_CaO	E_O	E_Ca	E_reaction (kcal/mol)	rel_error_abs (%)	E_dimer	E_monomer	E_interaction (kcal/mol)	rel_error_abs (%)
pbe	-752.553839	-75.014233	-677.343179	123.260263	28.195802	b3lyp	-159.72711	-79.864058	0.631584
b3lyp	-752.855416	-75.100403	-677.590183	103.432483	7.574005	b3lyp-d3(bj)	-159.741043	-79.889474	-1.314987
mn15	-752.768128	-75.058552	-677.555663	95.075000	0.182007	b3lyp-nl	-159.535328	-79.766805	-0.077809
mp2	-752.175898	-74.998951	-677.007943	106.049453	10.295843	pbe	-159.462022	-79.730878	-0.167299
scam	-752.860987	-75.073655	-677.61644	115.799546	20.436345	pbe-d3	-159.466307	-79.73184	-1.648583
reference				96.150000	0.000000	mp2	-159.298734	-79.648075	23.028547
						swvn	-158.825753	-79.41124	7.060562
						reference			53.292361
									0.000000
									0.000000

Figure 1: CaO

Figure 2: Ethane dimer

	E_eclipsed	E_staggered	E_reaction (kcal/mol)	rel_error_abs (%)
pbe	-115.627135	-115.628988	-1.162206	15.069940
b3lyp	-115.773822	-115.775579	-1.102359	9.144491
mn15	-115.63698	-115.638675	-1.063366	5.283808
mp2	-115.541834	-115.54374	-1.195833	18.399271
scam	-115.717814	-115.719749	-1.214093	20.207224
reference			-1.010000	0.000000

Figure 3: Methanol

Figure 4: Reaction energies and relative errors for the three test systems.

we need this dispersion dependency to better represent van der Waals interactions, and therefore our results are in agreement with b3lyp-d3(bj) being the preferred dispersion correction for most uses. Finally, CaO case gives results outside the chemical accuracy for all results except for mn15 giving an absolute error of 0.175 kcal/mol. Since MN15 is the only method that is almost exact for CaO, it is the most likely to have been parametrized on this reaction (Minnesota 2015B database). Our initial ranking is therfore confirmed.

Bonus Exercise 7

How could fortitous error cancellation help in getting good results when calculating reaction energies?

In practice, approximate density functionals often yield poor absolute total energies, but reasonably good reaction energies because of fortuitous error cancellation. If we write the approximate energies as $E^{\text{approx}} = E^{\text{true}} + \delta$, then for a reaction $R \rightarrow P$ we have

$$\Delta E^{\text{approx}} = (E_P^{\text{true}} + \delta_P) - (E_R^{\text{true}} + \delta_R) = \Delta E^{\text{true}} + (\delta_P - \delta_R).$$

When the systematic errors of reactants and products are similar ($\delta_P \approx \delta_R$), the difference $\delta_P - \delta_R$ is small and the reaction energy is much more accurate than the individual total energies. This is the same mechanism discussed in the lecture notes for the success of simple LDA-type functionals, where an overestimation of exchange and an underestimation of correlation partially cancel in E_{xc} .

Our results illustrate this behaviour. For the methanol conformers, the two structures are very similar electronically, so the DFA errors largely cancel and all methods achieve chemical accuracy ($|\Delta E| \leq \sim 0.25 \text{ kcal mol}^{-1}$), even though the underlying absolute energies are not exact. For the ethane dimer with functionals that lack long-range dispersion (e.g. plain B3LYP, PBE), the errors on the

dimer and on $2\times$ monomer are very different (missing $-C_6/R^6$ term only in the dimer), so error cancellation fails and the interaction energy is badly predicted. In the CaO dissociation, the bound state and the open-shell fragments are electronically very different, so most functionals (PBE, B3LYP, SCAN, MP2) show large BDE errors because their errors on CaO and on Ca + O do not cancel. MN15, which was parametrized on Minnesota 2015B data likely including CaO, is tuned so that these errors are more balanced, leading to an almost exact bond dissociation energy.

Exercise 8

What is the impact of using the dispersion correction for the ethane complex? For what phenomena are such corrections relevant? Give a real world example system.

For the ethane complex, adding a dispersion correction has a dramatic impact on the predicted interaction energy. Plain semi-local or hybrid DFAs such as B3LYP and PBE do not contain the long-range London term $-C_6/R^6$ and therefore give a very weakly bound or even slightly repulsive dimer (interaction energy close to 0 or positive), in clear disagreement with the reference value of $-1.34 \text{ kcal mol}^{-1}$. When we use dispersion-corrected variants such as b3lyp-d3(bj) or b3lyp-nl, which explicitly add either an empirical $-C_6/R^6$ potential (DFT-D3 with 2- and 3-body terms) or a non-local correlation functional of the VV10 type, the missing van der Waals attraction is restored and the interaction energy comes very close to the reference (chemical accuracy is achieved). This behaviour is exactly what is described in the notes: dispersion is a long-range correlation effect that standard local and semi-local functionals do not capture, and must be modeled either by non-local $v_c^{\text{nl}}(\mathbf{r})$ terms or by additive $-C_6/R^6$ corrections.

Such dispersion corrections are crucial whenever weak, noncovalent interactions and long-range correlation dominate the energetics: van der Waals complexes. A concrete real-world example is the benzene dimer: its binding energy and preferred stacked geometries are largely governed by dispersion, and functionals without an explicit dispersion treatment severely underestimate its stability, whereas dispersion-corrected DFAs and non-local functionals reproduce experimental data much more reliably. (benzene dimer)

Exercise 9

Which grid size should you use for this kind of problem? Are the fluctuations observed with the SG1 grid physical? Evaluate plots you create above.

For this type of weak van der Waals interaction Ar_2 , the SG1 grid is clearly insufficient. In both B3LYP and M06-HF plots the SG1 curve shows strong, irregular oscillations and even a spurious plateau at large R , which are numerical artefacts of the coarse integration grid rather than physical features of the potential energy surface. This indeed agrees with the notes, which tell us that PESs of non-bonded interactions (what we are dealing here) are particularly

sensitive to grid quality. The medium (75–302) and fine (99–590) grids give smooth, well behaved curves; for B3LYP the two almost coincide and converge to the correct behaviour of two independant Ar atoms, while for M06-HF the 99–590 grid still changes the depth and position of the minimum significantly. This method is demanding and its physical bias becomes more visible when the grid is improved (notably because of the kinetic energy density). For this kind of problem one should therefore at least use a 75–302 grid, and preferably a 99–590 grid for more demanding functionals or when accurate interaction energies are required. We also note that none of the methods and integration grids fully recover the reference curve and only reproduce the behavior on some portion of the graph. BUT, I ended up with one consideration, in the notebook we did this :

```

1 # Calculate energy of single Argon atom
2 psi4.core.clean_options()
3 ar = psi4.geometry("""
4 Ar 0 0 0
5 """)
6
7 E_ar = psi4.energy(functional+"/def2-QZVP", molecule = ar)

```

This forces to compute the energy of one atom of Argon with a unique grid and compare this single calculation to all different grids... Therefore I included the calculation of the Energy of one Ar in the loop over grids where we compute the interaction energy for different grids as following :

```

1 psi4.core.clean()
2 psi4.core.clean_options()
3 psi4.core.set_output_file('grid.log', False)
4 functional = "b3lyp" # b3lyp and m06-hf
5 E_atom = {}
6 energies = {}
7 distances = np.linspace(3, 6, 31)

8
9 grid_options = {
10     'sg1': {'dft_grid_name': 'sg1'},
11     '75-302': {'dft_radial_points': 75, 'dft_spherical_points': 302},
12     '99-590': {'dft_radial_points': 99, 'dft_spherical_points': 590}
13 }
14
15 for grid in grid_options.keys():
16     psi4.core.clean_options()
17     psi4.set_options(grid_options[grid])

18
19     ar = psi4.geometry("Ar 0 0 0")
20     E_atom[grid] = psi4.energy(functional+"/def2-QZVP", molecule=ar
21 )
22
23     energies[grid] = []
24
25     for R in distances:
26         # Argon Dimer Z-Matrix
27         ar_dimer = psi4.geometry(f"""
28             0 1
29             Ar

```

```

29         Ar 1 {R}
30         """)
31     energies[grid].append(psi4.energy(functional+"/def2-QZVP",
molecule=ar_dimer))

```

and then I compute the interaction energy grid to grid :

```

1 fig, ax = plt.subplots(1)
2 ax.set_title(functional)
3 # we plot the curves for each grid we used
4 for grid in grid_options.keys():
5     # the interaction energy is the E_dimer - 2*E_single
6     # you can use psi4.constants.hartree2kcalmol for the unit
7     # conversion
8     # you can access the energy for gridpoint X by using
9     # np.array(energies[X]), where X is the gridpoint you are
10    interested in
11    Interaction_Energy = (np.array(energies[grid]) - 2.0 * E_atom[
12        grid]) * psi4.constants.hartree2kcalmol
13    ax.plot(distances, Interaction_Energy, label=grid)
14 ax.plot(distances, reference, label='reference', linestyle="--")
15 ax.set_xlabel('Separation (A)')
16 ax.set_ylabel('Energy (kcal/mol)')
17 ax.legend()
18 fig.savefig(functional+'-integrationgrid.jpg')
19 plt.show()

```

instead of :

```

1 fig, ax = plt.subplots(1)
2 ax.set_title(functional)
3 # we plot the curves for each grid we used
4 for grid in grid_options.keys():
5     # the interaction energy is the E_dimer - 2*E_single
6     # you can use psi4.constants.hartree2kcalmol for the unit
7     # conversion
8     # you can access the energy for gridpoint X by using
9     # np.array(energies[X]), where X is the gridpoint you are
10    interested in
11    Interaction_Energy = (np.array(energies[grid]) - 2.0 * E_ar) *
12        psi4.constants.hartree2kcalmol
13    ax.plot(distances, Interaction_Energy, label=grid)
14 ax.plot(distances, reference, label='reference', linestyle="--")
15 ax.set_xlabel('Separation (A)')
16 ax.set_ylabel('Energy (kcal/mol)')
17 ax.legend()
18 fig.savefig(functional+'-integrationgrid.jpg')
19 plt.show()

```

And we seem to endup with more reliable results as shown in Figure 9 for M06-HF but not B3LYP.

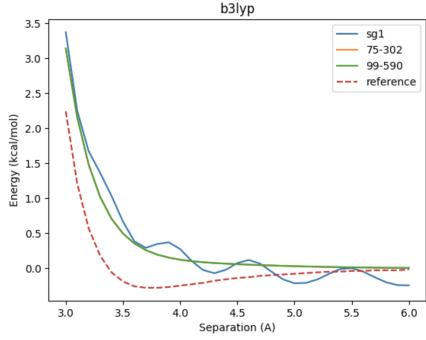


Figure 5: Interaction energy calculated with $b3lyp$ for different integration grids

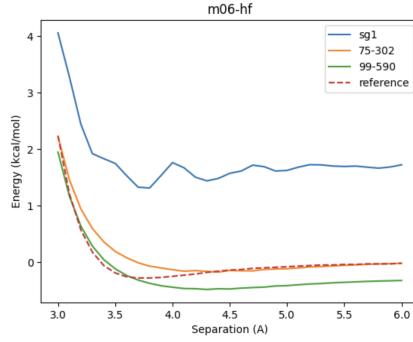


Figure 6: Interaction energy calculated with $m06\text{-hf}$ for different integration grids

Figure 7: Interaction energy as a function of the separation of Ar_2

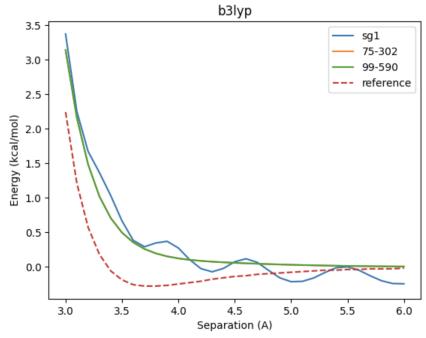


Figure 8: Interaction energy calculated with $b3lyp$ for different integration grids

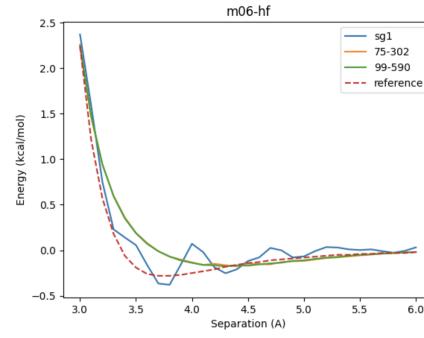


Figure 9: Interaction energy calculated with $m06\text{-hf}$ for same integration grids

Figure 10: Interaction energy as a function of the separation of Ar_2 with grid to grid comparison

Exercise 10

Compare the two functionals regarding convergence with respect to the integration grid. Which functional shows concerning behavior in this regard? Evaluate plots you create below.

If we consider the case in which we compute E_{Ar} with a unique grid : B3LYP shows reasonably good convergence with respect to the integration grid where the 75–302 and 99–590 curves are very similar, and only the SG1 grid produces visible noise. In contrast, M06-HF is much more sensitive to the grid. The

SG1 curve is qualitatively wrong and highly oscillatory, and even the difference between 75–302 and 99–590 is substantial with different well depth and shape. This reflects the greater grid demands of modern meta-GGA hybrids that depend on the kinetic-energy density. Thus, M06-HF shows the most concerning behaviour with respect to integration-grid convergence. On the other hand B3LYP is relatively robust once a medium grid is used.