

Homework 4: Bound states in the central potential II

Solve the multielectron atom in LDA approximation.

- Test it on He and oxygen by computing the total energy and charge density
- Plot charge density
- Print the total energy and compare it to the database at NIST

<https://www.nist.gov/pml/data/atomic-total-energies-and-ei>

The algorithm is sketched below. Fell free to do it in your own way.

We want to solve the Schroedinger equation for an atom with nucleous charge Z . We will approximate electron-electron interaction with an effective potential, which is computed by so-called "local density approximation" (LDA). In this theory, the classical (Hartree) potential is treated exactly, while the rest of the interaction is "hidden" into approximate exchange-correlation functional. We will not go into details of this functional, we will just use

it here.

The Schroedinger equation we are solving is

$$\left[-\nabla^2 - \frac{2Z}{|\mathbf{r}|} + V_H(\mathbf{r}) + V_{xc}(\mathbf{r})\right]\psi(\mathbf{r}) = \varepsilon\psi(\mathbf{r}) \quad (1)$$

The first two terms are appearing in Hydrogen problem, and we already coded them.

The Hartree term is treated exactly in this approximation. It describes the electrostatic interaction of one electron with the cloud of all electrons (including the electron itself).

Mathematically, this term is

$$\begin{aligned} & \frac{1}{2} \int d\mathbf{r}d\mathbf{r}' \psi^\dagger(\mathbf{r})\psi^\dagger(\mathbf{r}')v_c(\mathbf{r} - \mathbf{r}')\psi(\mathbf{r}')\psi(\mathbf{r}) \rightarrow \\ & \int d\mathbf{r}\psi^\dagger(\mathbf{r})\psi(\mathbf{r}) \int d\mathbf{r}' \langle \psi^\dagger(\mathbf{r}')\psi(\mathbf{r}') \rangle v_c(\mathbf{r} - \mathbf{r}') \equiv \int d\mathbf{r}\psi^\dagger(\mathbf{r})V_H(\mathbf{r})\psi(\mathbf{r}) \end{aligned} \quad (2)$$

with

$$V_H(\mathbf{r}) = 2 \int d\mathbf{r}' \frac{\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} \quad (3)$$

where 2 is due to Rydberg units sinc $v_c = 2/r$.

For any atom, the electron density is spherically symmetric and hence V_H depends only on

radial distance. (In solids, the hartree potential should be expanded in spheric harmonics to sufficiently high l , maybe $l = 6$).

- Step 1: Using $\rho(r)$ computed in previous homework, compute the Hartree potential.

This is usually achieved by solving the Poisson equation. From classical electrostatic we know

$$\nabla^2 V_H(\mathbf{r}) = -8\pi\rho(\mathbf{r}) \quad (4)$$

In Hartree approximation, we have

$$\frac{1}{r^2} \frac{d}{dr} \left(r^2 \frac{dV_H}{dr} \right) = -8\pi\rho(r) \quad (5)$$

which simplifies to

$$U''(r) = -8\pi r \rho(r) \quad (6)$$

where $U(r) = V_H(r)r$.

This second order differential equation has the following boundary conditions $U(0) = 0$ and $U(\infty) = 2Z$.

The two point boundary problem does not require shooting because we know solution to ~~the homogenous differential equation $U''(r) = 0$. The Hartree potential can be obtained~~

from any particular solution by

$$U(r) = U_p(r) + \alpha r \quad (7)$$

where $\alpha = \lim_{r \rightarrow \infty} (2Z - U_p(r))/r$.

- Step 2: Compute the exchange correlation potential.

Note that $V_{xc}(\mathbf{r}) = V_{xc}(\rho(\mathbf{r}))$ is uniquely determined by electron charge density $\rho(\mathbf{r})$. If you know ρ , you can compute V_{xc} by the module provided.

Download the module "excor.py" from

http://www.physics.rutgers.edu/~haule/509/src_prog/python/homework5/

and import it in your code.

Instantiate the ExchangeCorrelation object by

```
exc = ExchangeCorrelation()
```

and used it, for example, by

```
exc.Vx(rs(rho[i]))+exc.Vc(rs(rho[i]))
```

where $r_s = (4\pi\rho/3)^{-1/3}$. Be careful: The energy unit in "excor.py" is Hartree and not Rydergs. Hence, you need to multiply energy or potential by 2.

- Step 3: Add the Hartree potential and the exchange-correlation potential to the Schroedinger equation and find bound states of the atom.

The Schroedinger equation is

$$-u''(r) + \left(\frac{l(l+1)}{r^2} - \frac{2Z}{r} + V_H(r) + V_{XC}(r) \right) u(r) = \varepsilon u(r). \quad (8)$$

- Step 4: Compute the new electron density by filling the lowest Z eigenstates.
- Step 5: Admix the new density to the old density (50% of the old and 50% of the new should work) and use the resulting density to compute the new Hartree and exchange-correlation potential.
- Iterate above steps until self-consistency is achieved.

Once you see that the code converges, you should insert a new step between *Step 4* and *Step 5* to compute total energy of the system. The total energy can be obtained by

$$E_{total}^{LDA} = \sum_{i \in occupied} \int d\mathbf{r} \psi_i^*(\mathbf{r}) [-\nabla^2] \psi_i(\mathbf{r}) + \int d\mathbf{r} \rho(\mathbf{r}) [V_{nucleous}(\mathbf{r}) + \epsilon_H(\mathbf{r}) + \epsilon_{XC}(\mathbf{r})]$$

$$\begin{aligned}
&= \sum_{i \in \text{occupied}} \int d\mathbf{r} \psi_i^*(\mathbf{r}) [-\nabla^2 + V_{\text{nucleous}} + V_H + V_{XC}] \psi_i(\mathbf{r}) \\
&+ \int d\mathbf{r} \rho(\mathbf{r}) [\epsilon_H(\mathbf{r}) - V_H(\mathbf{r}) + \epsilon_{XC}(\mathbf{r}) - V_{XC}(\mathbf{r})] \\
&= \sum_{i \in \text{occupied}} \epsilon_i + \int d\mathbf{r} \rho(\mathbf{r}) [\epsilon_H(\mathbf{r}) - V_H(\mathbf{r}) + \epsilon_{XC}(\mathbf{r}) - V_{XC}(\mathbf{r})] \\
&= \sum_{i \in \text{occupied}} \epsilon_i + \int d\mathbf{r} \rho(\mathbf{r}) [-\epsilon_H(\mathbf{r}) + \epsilon_{XC}(\mathbf{r}) - V_{XC}(\mathbf{r})] \quad (9)
\end{aligned}$$

Here we used

$$E_y[\rho] \equiv \int d\mathbf{r} \rho(\mathbf{r}) \epsilon_y[\rho(\mathbf{r})] \quad (10)$$

$$V_y[\rho] \equiv \frac{\delta E_y[\rho]}{\delta \rho(\mathbf{r})} \quad (11)$$

where y is one of H , x or c .

Hence

$$\epsilon_H(\mathbf{r}) = \frac{1}{2}V_H(\mathbf{r}) = \frac{1}{2} \frac{U_H(\mathbf{r})}{r} \quad (12)$$

because

$$E_H = \int d\mathbf{r}d\mathbf{r}' \frac{\rho(\mathbf{r})\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} \quad (13)$$

The exchange-correlation energy can be obtained by a call to the method of *ExchangeCorrelation* object.

Compare your results with known values from:

<http://physics.nist.gov/PhysRefData/DFTdata/Tables/ptable.html>

GOOD LUCK!