

# Born-Oppenheimer Approximation

potential energy  
of the molecule

quantum energy  
of the electron

$$U = \mathcal{E} + \frac{e^2}{4\pi\epsilon_0 R}$$

classical energy of  
the protons

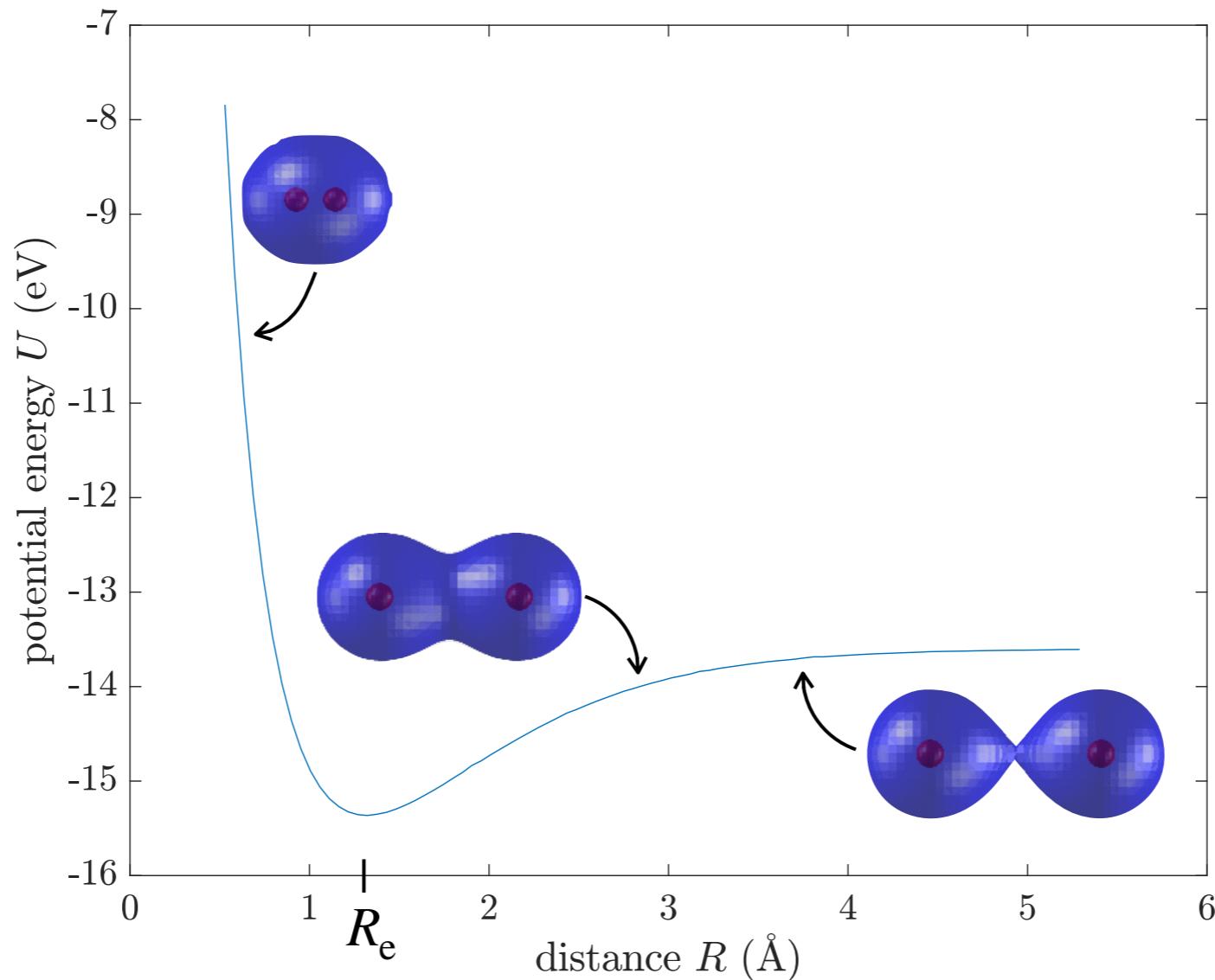


Max Born  
(1882-1970)



Julius Robert  
Oppenheimer  
(1904-1967)

# One-electron system



Max Born  
(1882-1970)



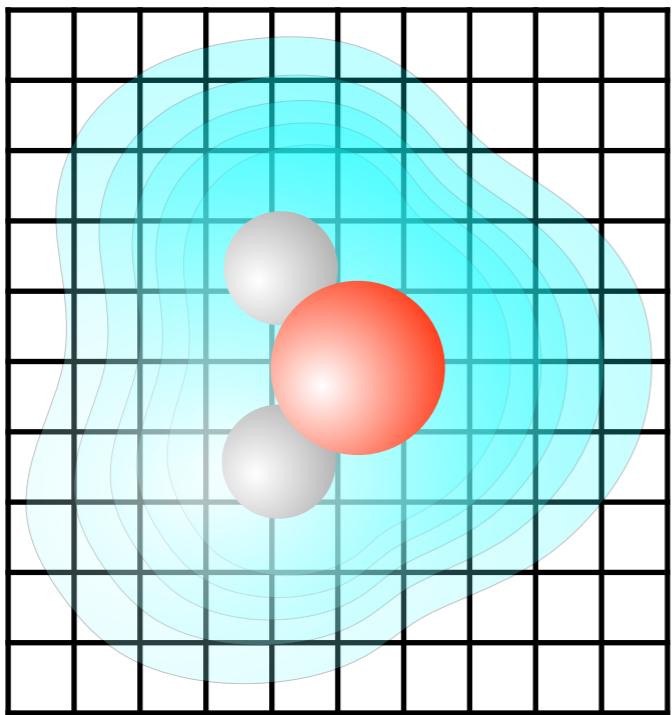
Julius Robert  
Oppenheimer  
(1904-1967)

# Many-electron system

---

- ▶ Solving the **many-electron problem** requires considerable computer memory and CPUs

Let us assume for example that we want to store the 10-electron wave function of our water molecule in the computer on a relatively coarse grid of  $10 \times 10 \times 10$  points

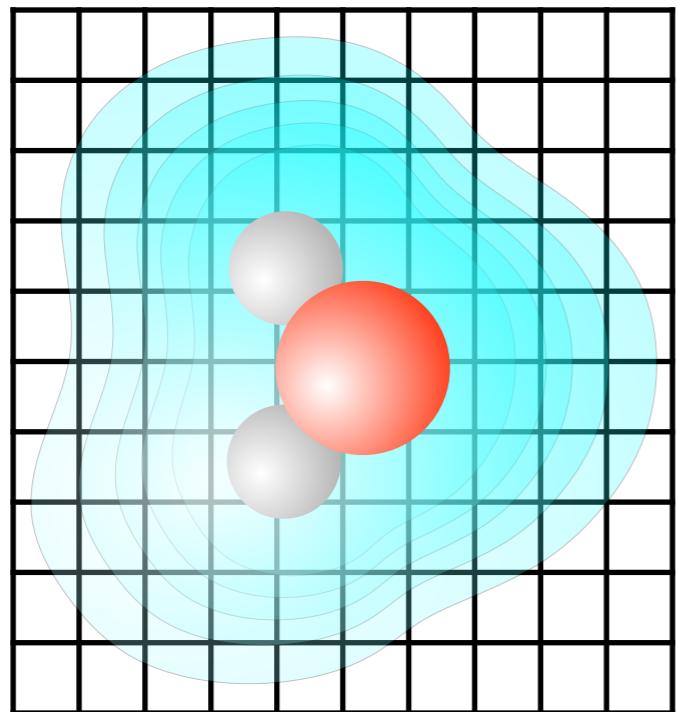


$$\Psi(\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_{10})$$

# Exponential bottleneck

---

- ▶ Solving the **many-electron problem** requires considerable computer memory and CPUs



Let us assume for example that we want to store the 10-electron wave function of our water molecule in the computer on a relatively coarse grid of  $10 \times 10 \times 10$  points

$$\Psi(\underbrace{\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_{10}}_{10^3}, \underbrace{\mathbf{r}_{10}}_{10^3})$$

We would need to store  $10^{30}$  complex number. If each complex number is 8 bytes, we get  $\sim 10^{19}$  Tb, which exceed the memory of all computers in the world ( $\sim 100\text{-}1,000$  Eb =  $10^8\text{-}10^9$  Tb)

- ▶ This example illustrates why the many-electron problem **cannot be solved by brute force**

$$\overset{\text{many-body}}{\curvearrowleft} H\Psi = E\Psi$$

# Density-functional theory

---

$$H\Psi = E\Psi$$

many-body

$$h_\rho \varphi = \varepsilon \varphi$$

one-body/effective-field  
(density-dependent)

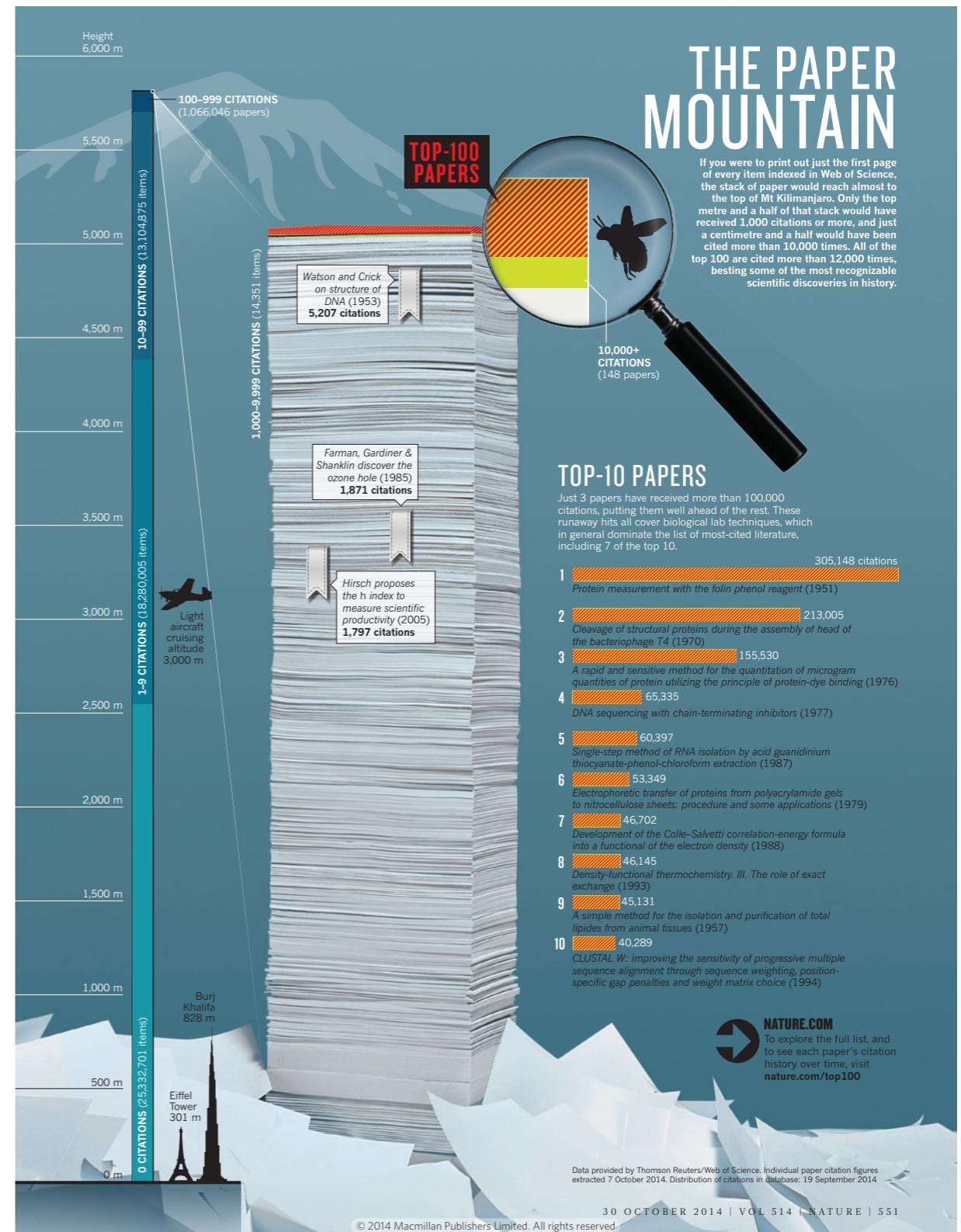
# Density-functional theory

$$H\Psi = E\Psi$$

many-body

$$h_\rho \varphi = \varepsilon \varphi$$

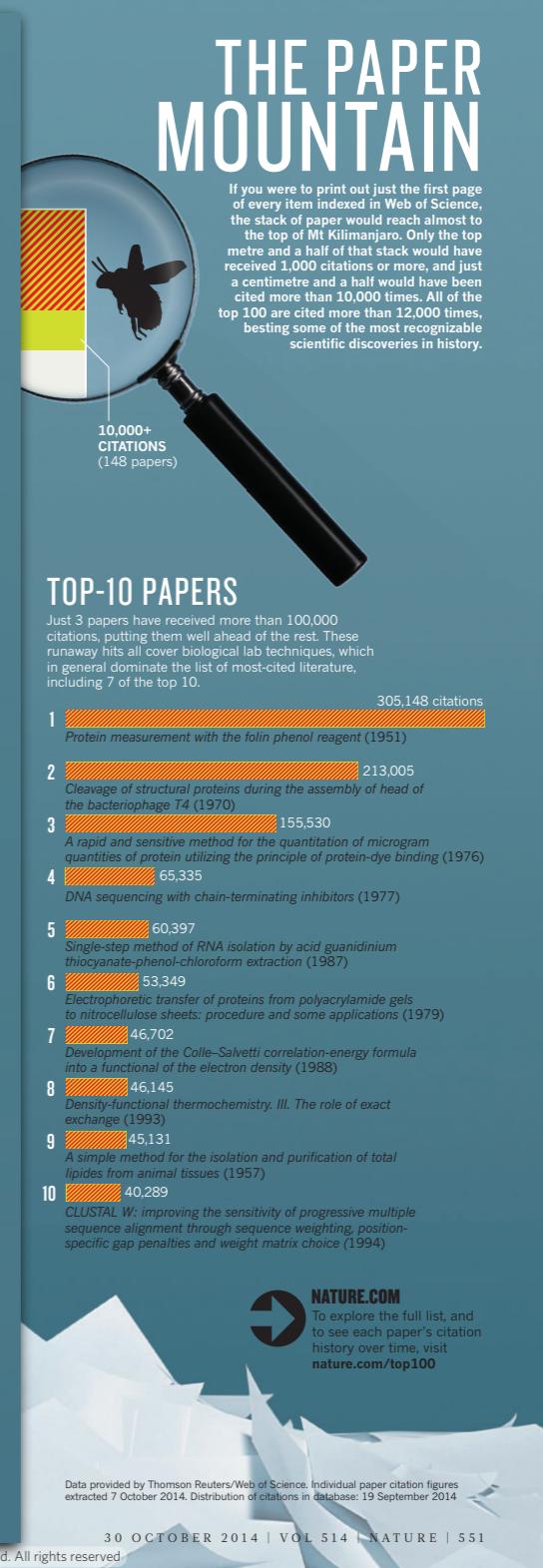
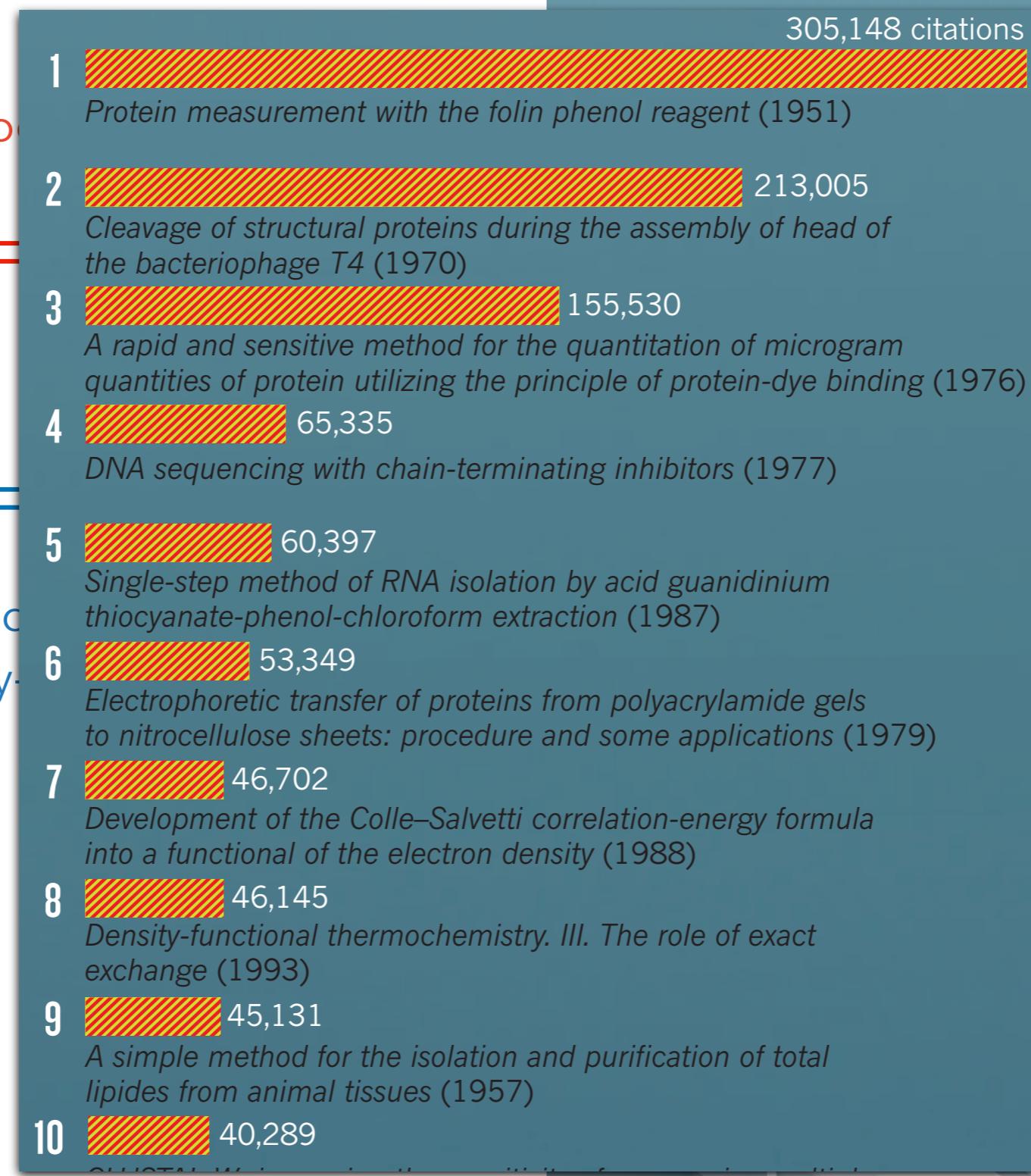
one-body/effective-field  
(density-dependent)



# Density-functional theory

$H\Psi = h_p \varphi =$

many-body  
one-body  
(density)



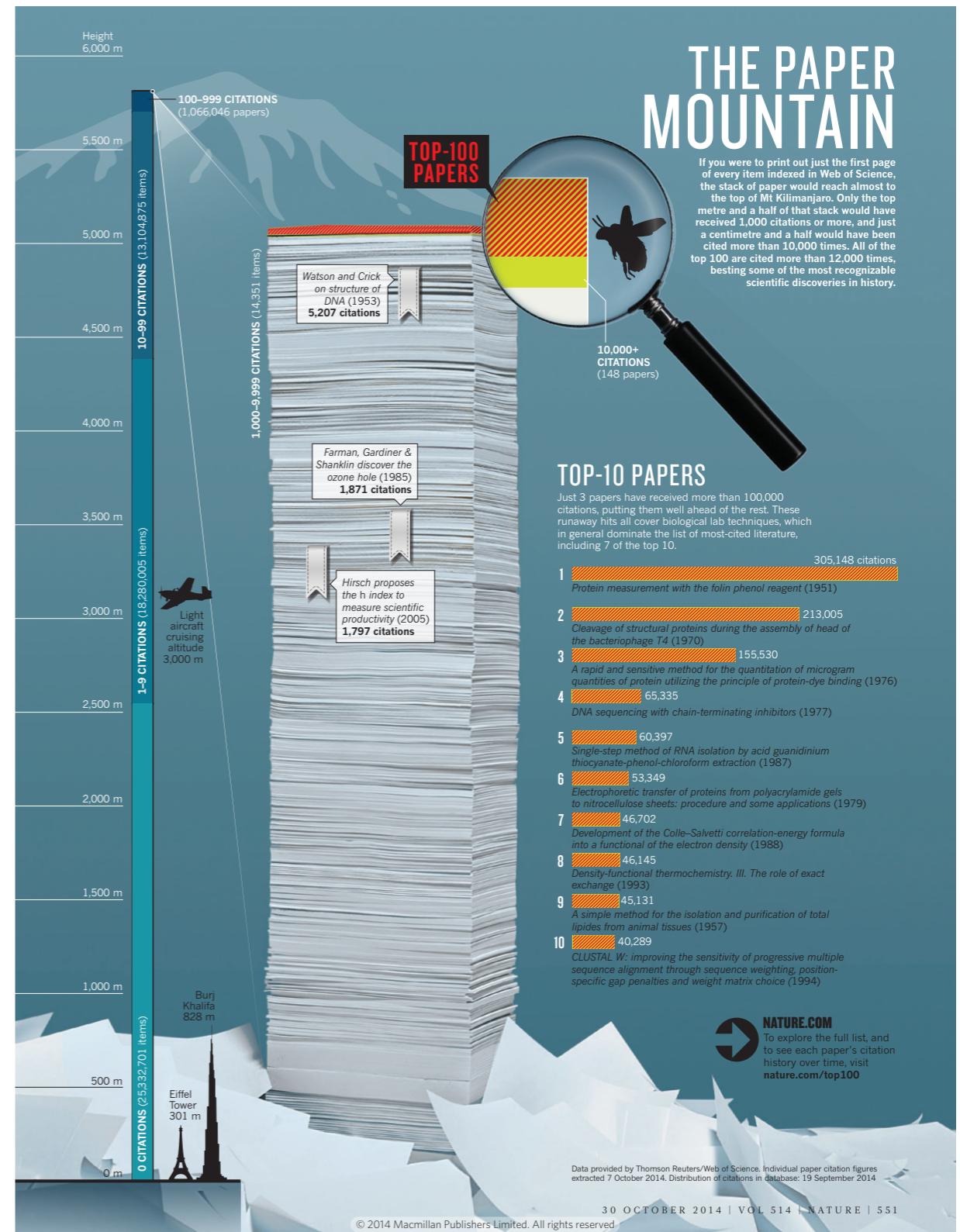
# Density-functional theory

$$H\Psi = E\Psi$$

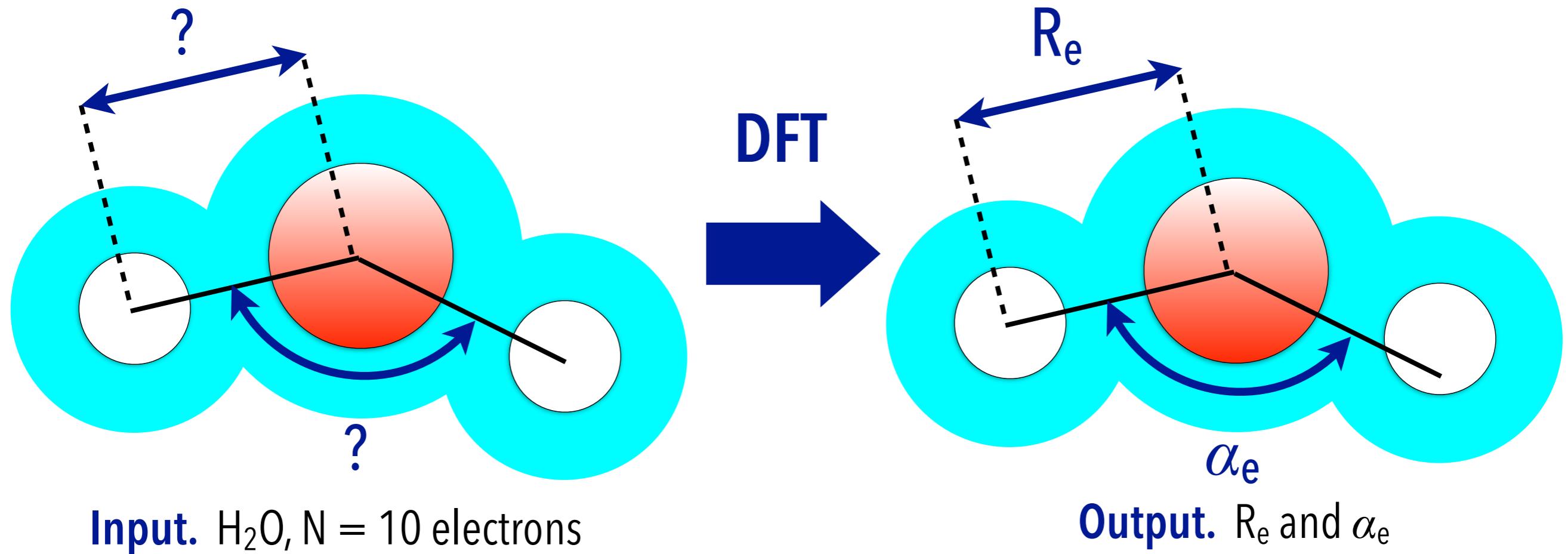
many-body

$$h_{\rho} \varphi = \varepsilon \varphi$$

one-body/effective-field  
(density-dependent)



# Density-functional theory



**Input.**  $\text{H}_2\text{O}$ ,  $N = 10$  electrons

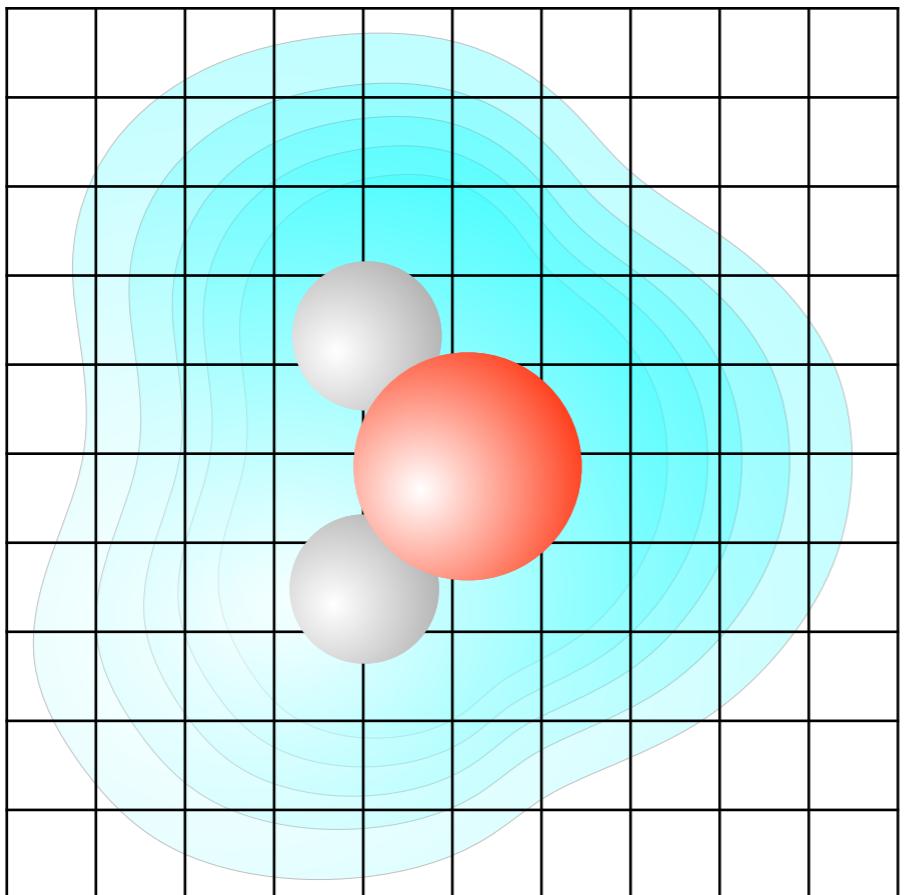
**Output.**  $R_e$  and  $\alpha_e$

**Density-Functional Theory (DFT)** is a quantum method widely used in materials science and chemistry to calculate the properties of systems of atoms (energy, geometry, reactivity, phase stability) using limited experimental data.

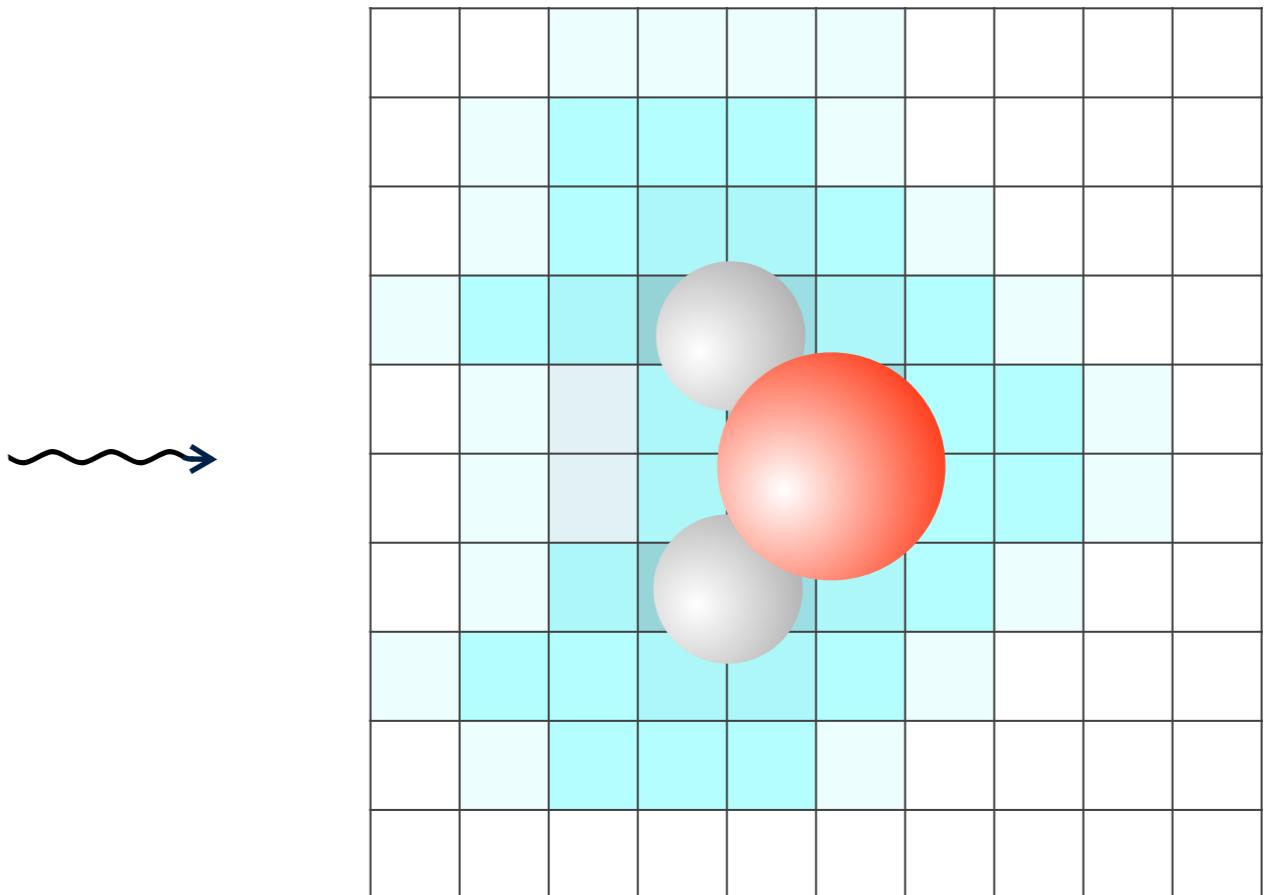
# Uniform electron gas (cont'd)

---

- Now, let's see how to approximate  $T[\rho]$ ,  $E_x[\rho]$ , and  $E_c[\rho]$  using a **uniform electron gas** as a model:



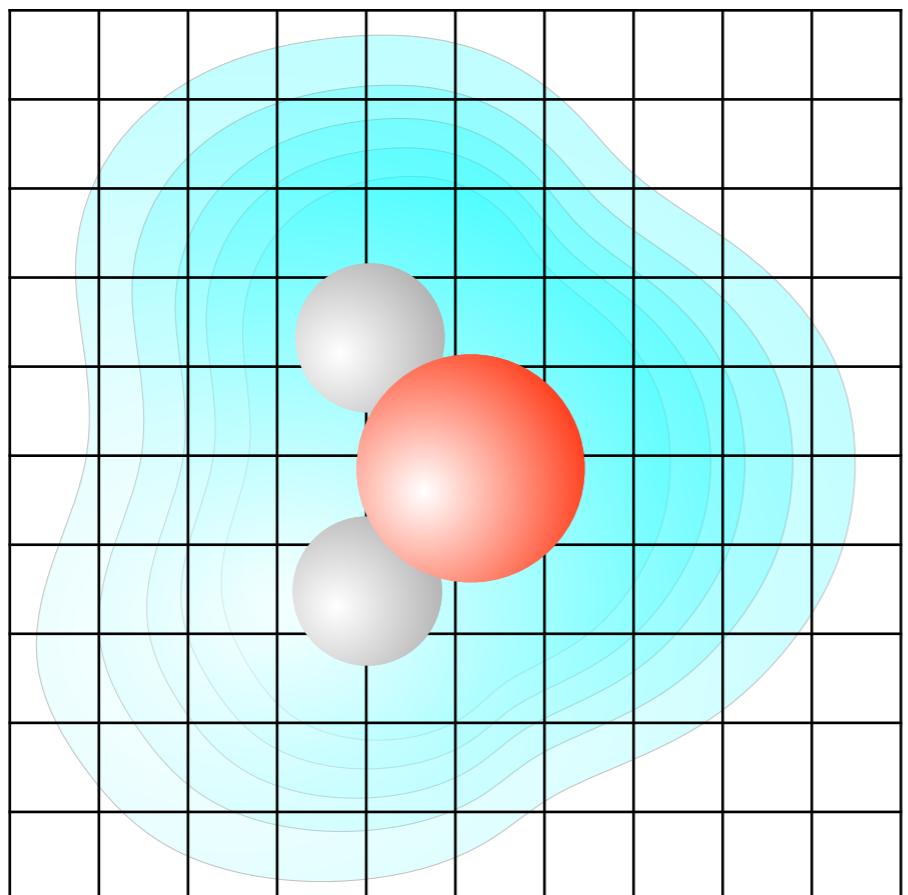
... Divide space into small volumes



... In each of these small volumes,  
approximate the density as uniform

# Uniform electron gas (cont'd)

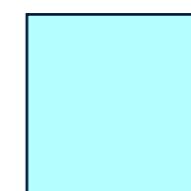
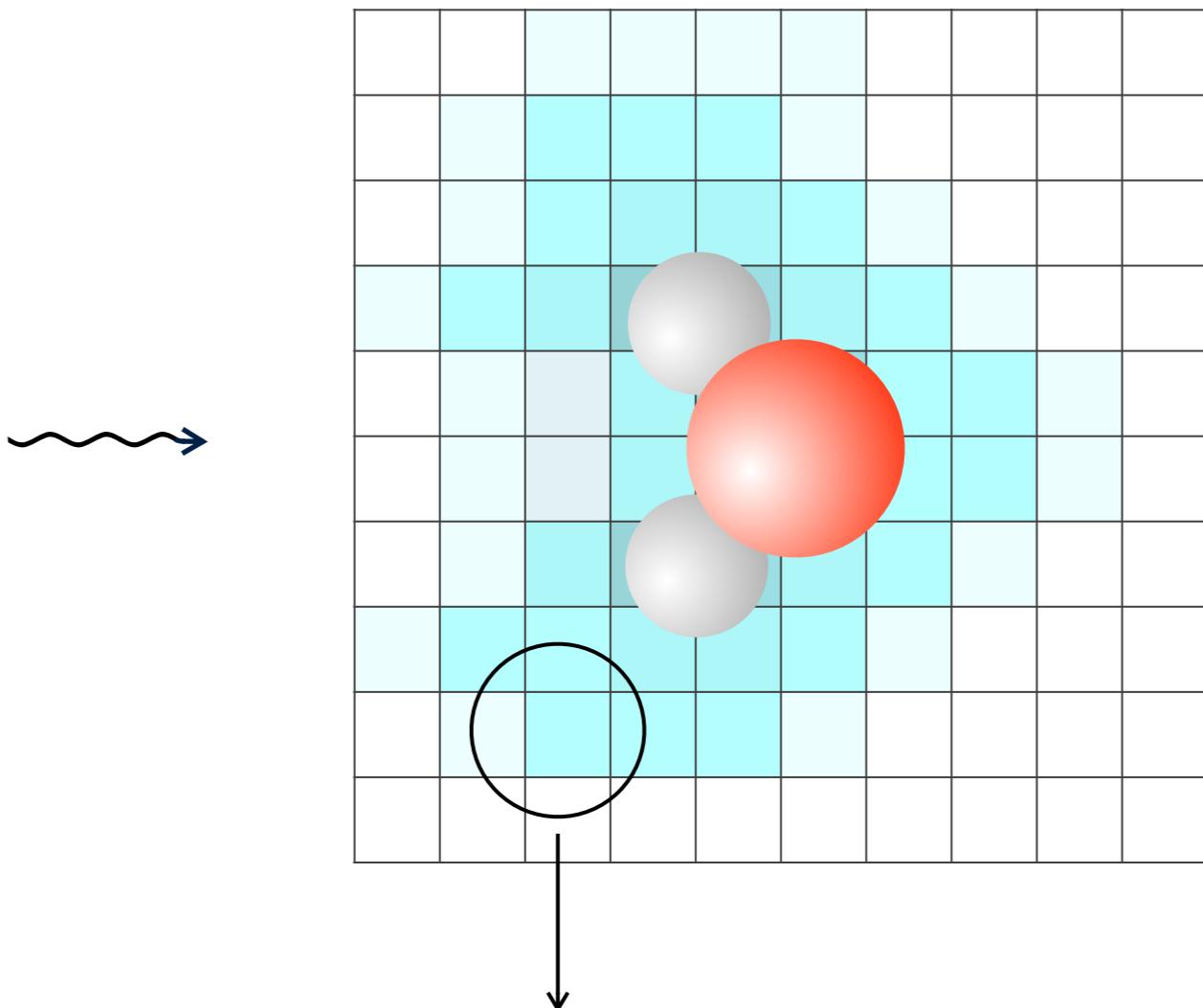
- Now, let's see how to approximate  $T[\rho]$ ,  $E_x[\rho]$ , and  $E_c[\rho]$  using a **uniform electron gas** as a model:



Kinetic energy per volume  
in uniform gas of density  $\rho$

$$dT = \underbrace{t(\rho(\mathbf{r}))}_{\text{Kinetic energy contribution}} d\mathbf{r}$$

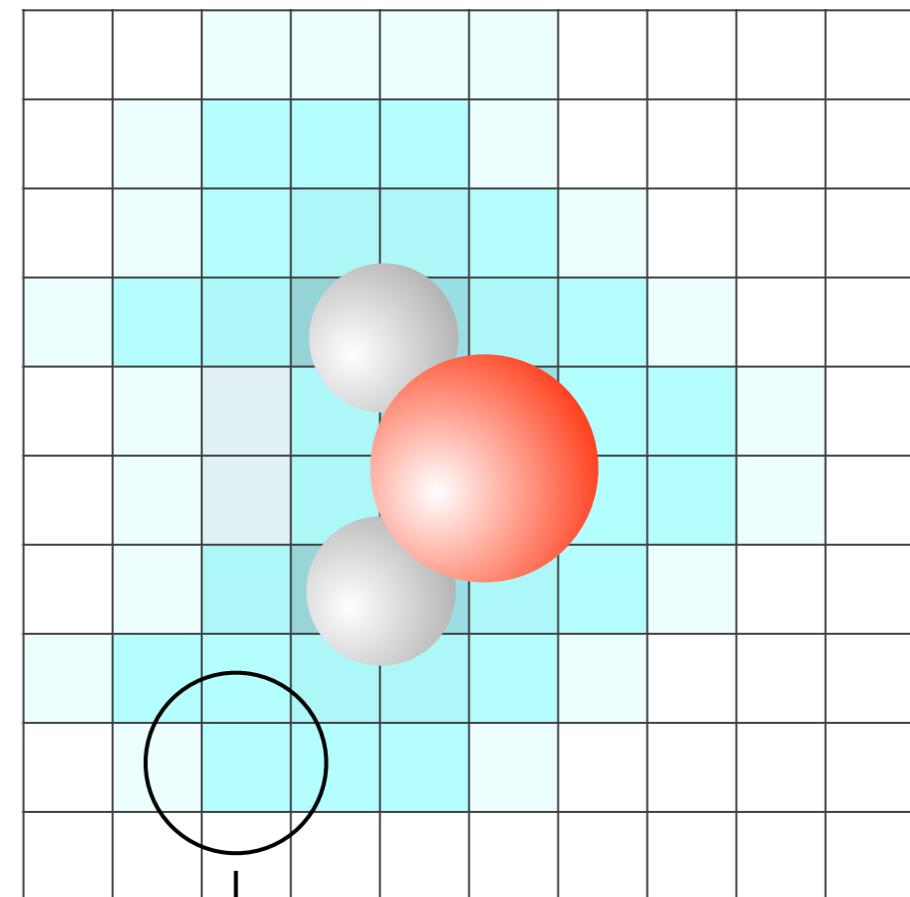
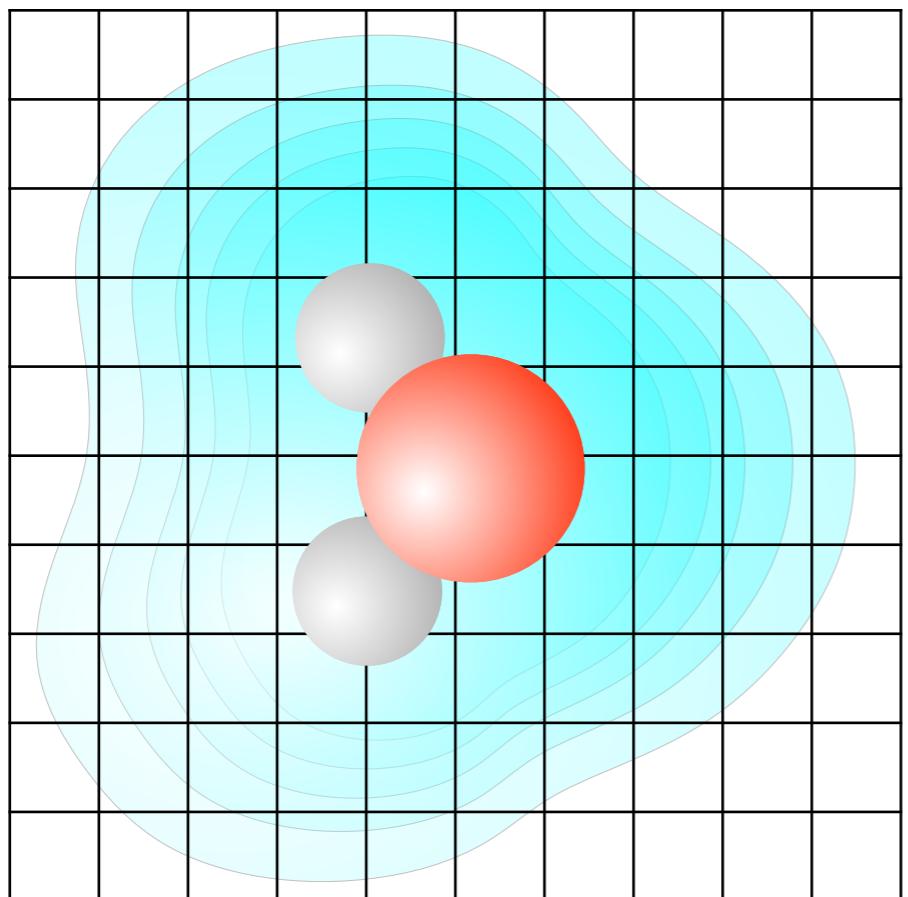
Kinetic energy contribution  
from the small volume



... Use uniform electron gas  
approximation to estimate  
 $T[\rho]$  in small volume

# Uniform electron gas (cont'd)

- Now, let's see how to approximate  $T[\rho]$ ,  $E_x[\rho]$ , and  $E_c[\rho]$  using a **uniform electron gas** as a model:



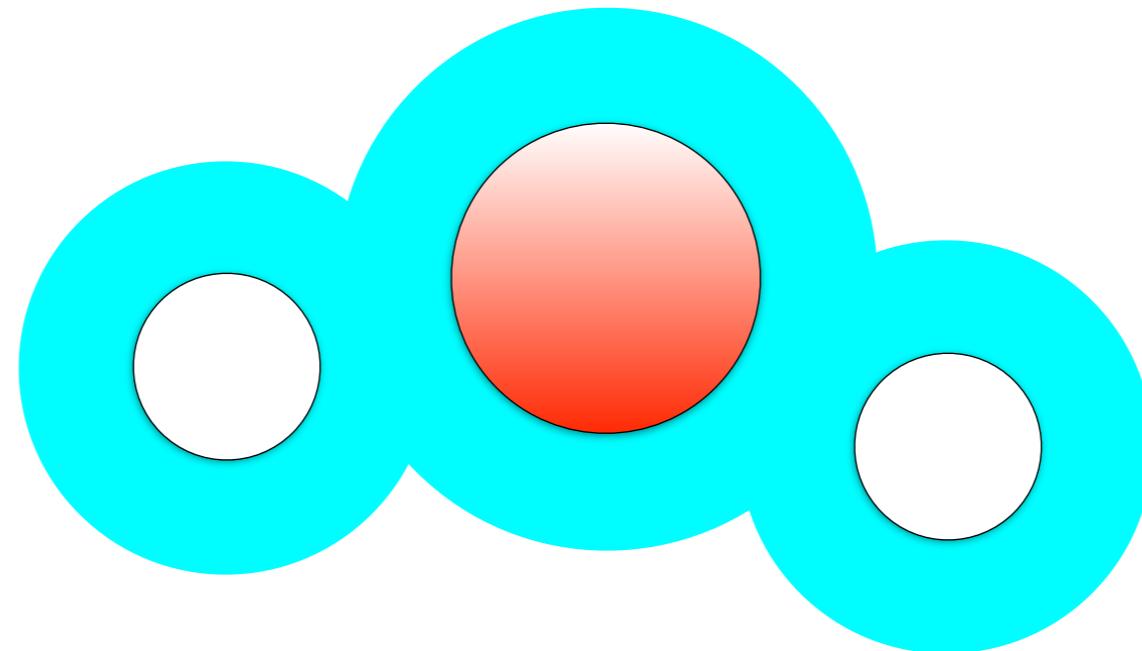
... And the same procedure can be applied to  $E_x[\rho]$  and  $E_c[\rho]$

$$dT = t(\rho(\mathbf{r})) d\mathbf{r}$$

$$dE_x = e_x(\rho(\mathbf{r})) d\mathbf{r}$$

$$dE_c = e_c(\rho(\mathbf{r})) d\mathbf{r}$$

# Density-functional theory (cont'd)



$$\begin{aligned} U = & \sum_n \int d\mathbf{r} \psi_n^*(\mathbf{r}) \cdot \mathcal{H} \psi_n(\mathbf{r}) \\ & + \frac{1}{2} \int d\mathbf{r} \int d\mathbf{r}' \frac{e^2 \rho(\mathbf{r}) \rho(\mathbf{r}')}{4\pi\epsilon_0 |\mathbf{r} - \mathbf{r}'|} + E_{xc}(\rho) \\ & + \frac{1}{2} \sum_i \sum_j' \frac{Z_i Z_j e^2}{4\pi\epsilon R_{ij}} \end{aligned}$$

# Minimization

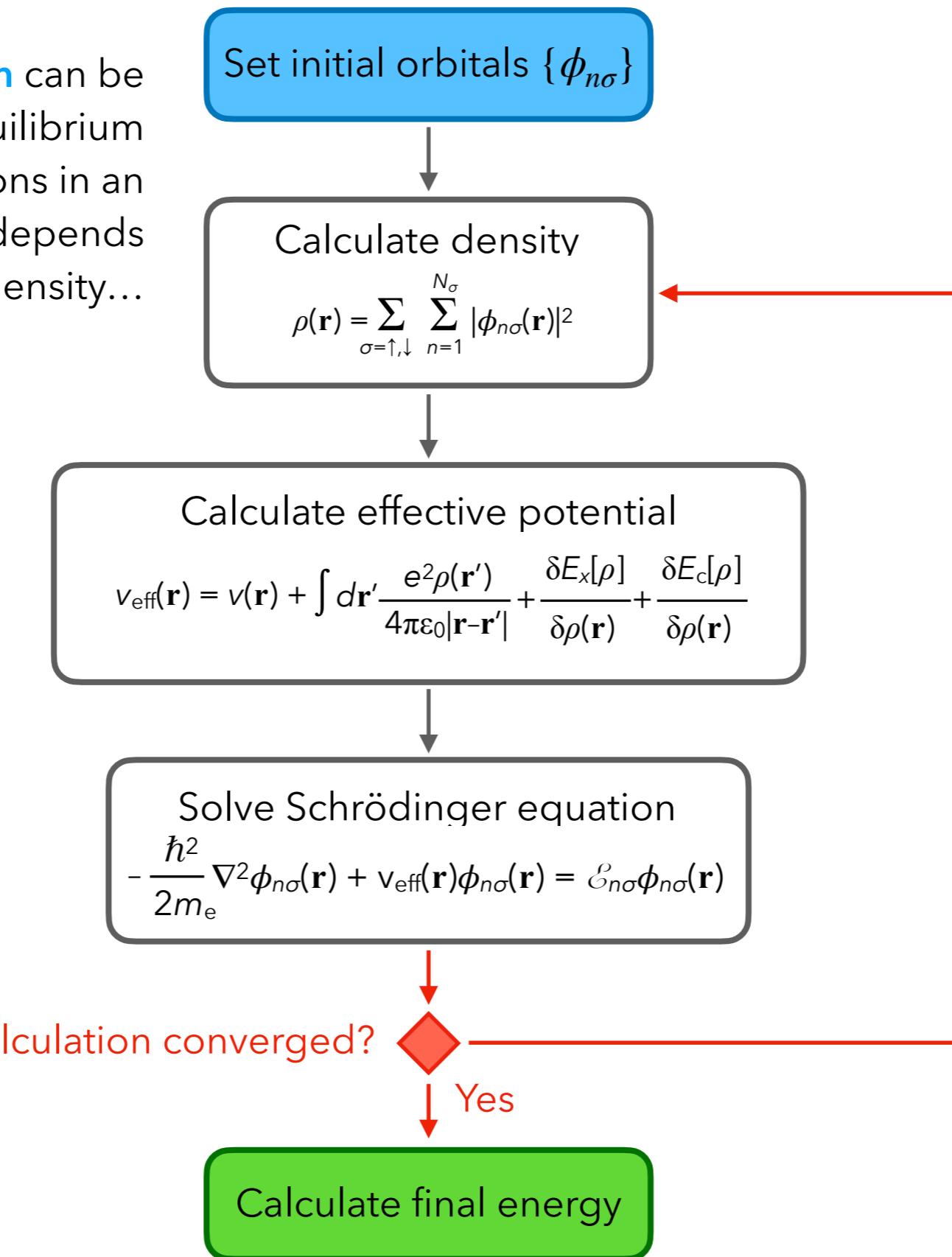
---

# Application

---

# Kohn-Sham self-consistent-field (SCF) cycle

The **Kohn-Sham problem** can be solved by finding the equilibrium states of the electrons in an effective potential that depends on the density...



Walter Kohn  
(1923-2016)

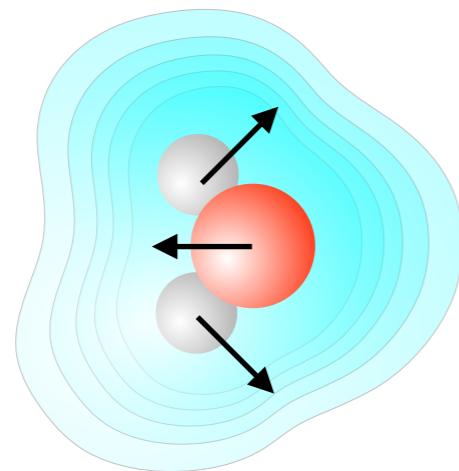


Lu Sham  
(born 1938)

# How to perform geometry optimization

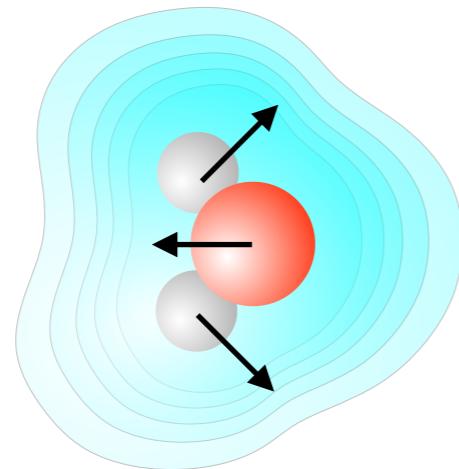
---

- Once the ground state density is found it is possible to calculate the forces on the nuclei and optimize the geometry of the system

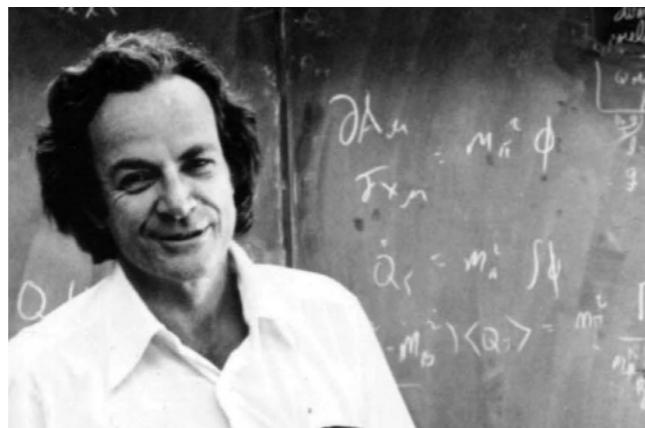


# How to perform geometry optimization

- Once the ground state density is found it is possible to calculate the forces on the nuclei and optimize the geometry of the system using the **Hellmann-Feynman theorem**



$$\mathbf{F}_i = - \int d\mathbf{r} \frac{e^2 Z_i \rho(\mathbf{r})}{4\pi\epsilon_0 |\mathbf{R}_i - \mathbf{r}|^2} \frac{(\mathbf{R}_i - \mathbf{r})}{|\mathbf{R}_i - \mathbf{r}|} + \sum_j' \frac{Z_i Z_j e^2}{4\pi\epsilon_0 R_{ij}^2} \frac{\mathbf{R}_{ij}}{R_{ij}}$$

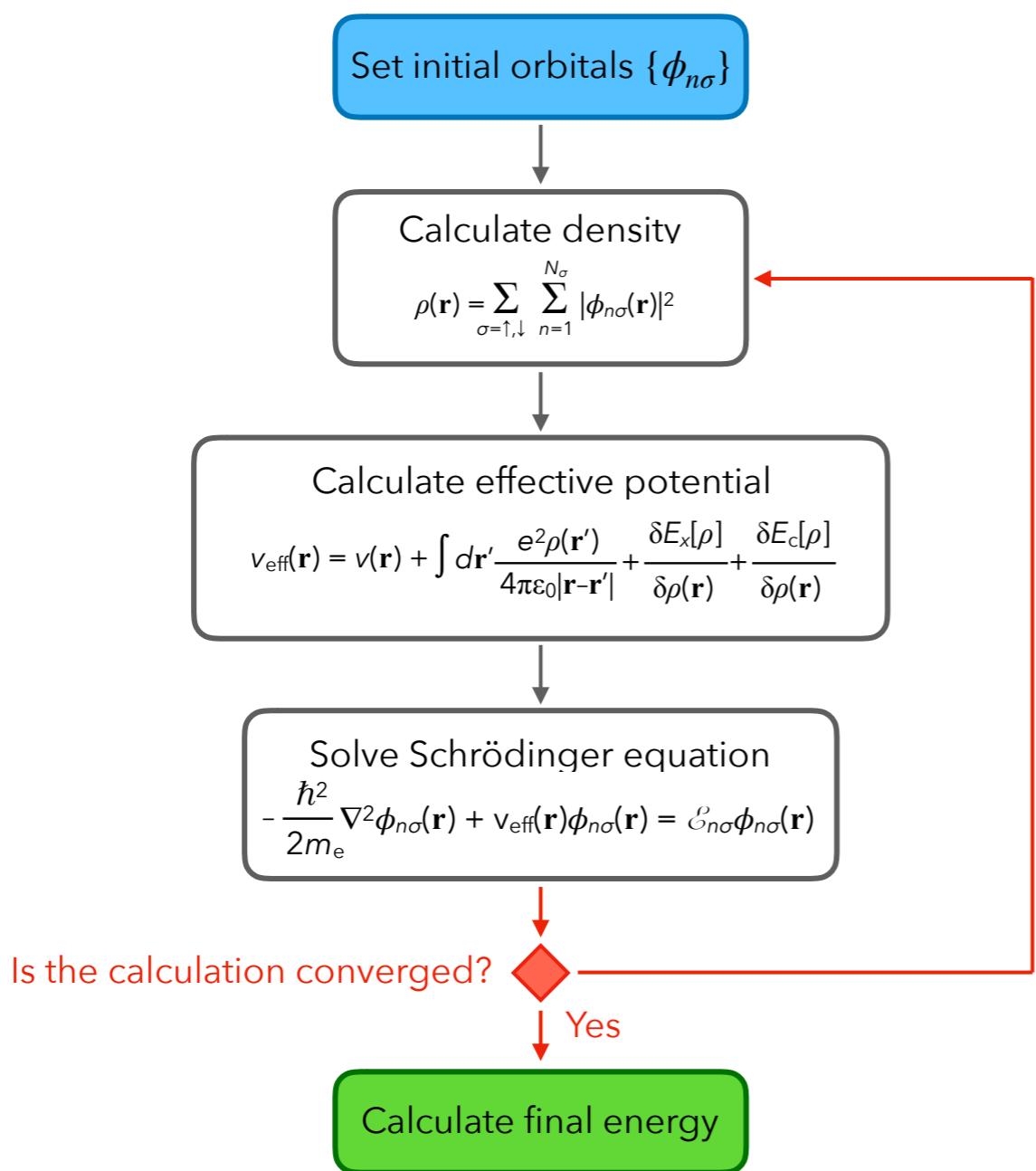


The first term is the attractive electrostatic force from the negative electrons while the second term is the repulsive electrostatic force from the other positive nuclei.

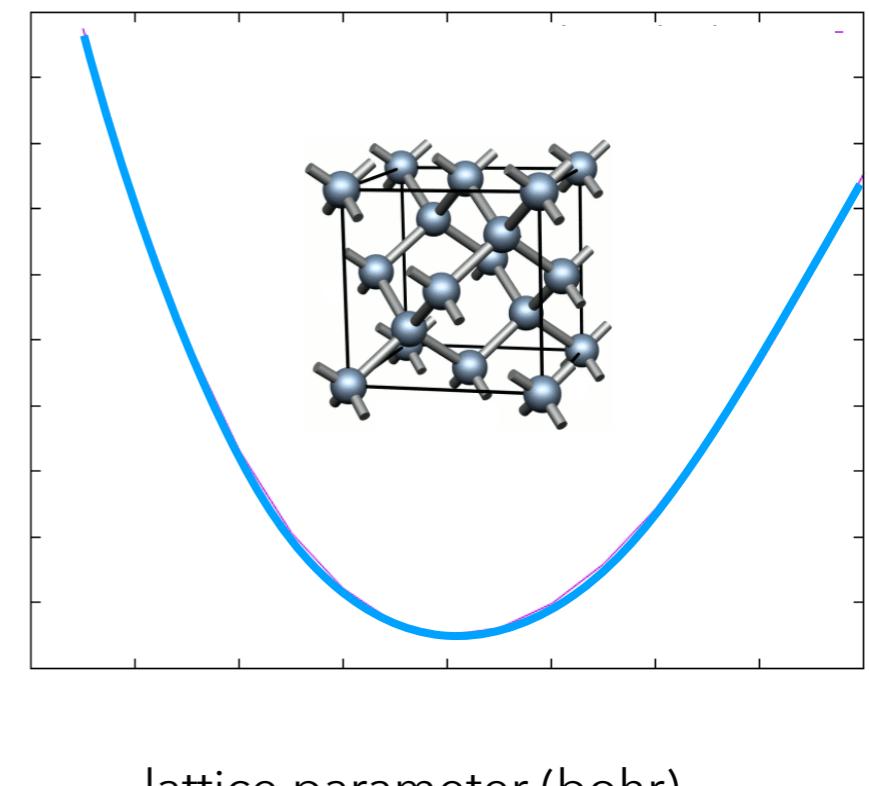
Richard Feynman (1918-1988)

# Preparing for our next laboratory

- We will learn how to predict the properties of materials using **density-functional theory** software



We will learn how to perform Kohn-Sham calculations using a software called **Quantum-Espresso**. There exist many other software ABINIT, VASP, CASTEP. They are all based on similar algorithms and input files.



# Numerical accuracy

- Importantly, we will need to tune two parameters of numerical accuracy, namely, the **cutoff kinetic energy** and the **number of Bloch wave vectors**.

Input cards '&... /' for a **Quantum-Espresso** simulation

```
&system
  ibrav = 2,
  celldm(1) = 10.2,
  nat = 2,           Plane-wave cutoff
  ntyp = 1,          ↘ kinetic energy
  ecutwfc = 30.0
/
&electrons
/
ATOMIC_SPECIES
Si 28.086 Si.pz-vbc.UPF
ATOMIC_POSITIONS alat
Si 0.00 0.00 0.00
Si 0.25 0.25 0.25
KPOINTS automatic    Number of k-points
                      to sample the Brillouin zone
                      4 4 4 1 1 1
```

$$\phi_{n\mathbf{k}}(\mathbf{r}) = \sum_{|\mathbf{G}| < G_{\text{cutoff}}} c_{n\mathbf{k},\mathbf{G}} e^{i(\mathbf{k}+\mathbf{G}) \cdot \mathbf{r}}$$

$$\rho(\mathbf{r}) = \frac{1}{N_{\mathbf{k}}} \sum_{\mathbf{k}} \sum_n |\phi_{n\mathbf{k}}(\mathbf{r})|^2$$

# Basis set: Guaussian vs. plane wave

---

# Numerical accuracy

---

## Plane-wave cutoff

- The orbitals of periodic systems can be written as a superposition of plane waves (Bloch's theorem)

$$\phi_{n\mathbf{k}}(\mathbf{r}) = \sum_{|\mathbf{G}| < G_{\text{cutoff}}} c_{n\mathbf{k},\mathbf{G}} e^{i(\mathbf{k}+\mathbf{G}) \cdot \mathbf{r}}$$

Bloch vector in the Brillouin zone

Vector of the reciprocal lattice

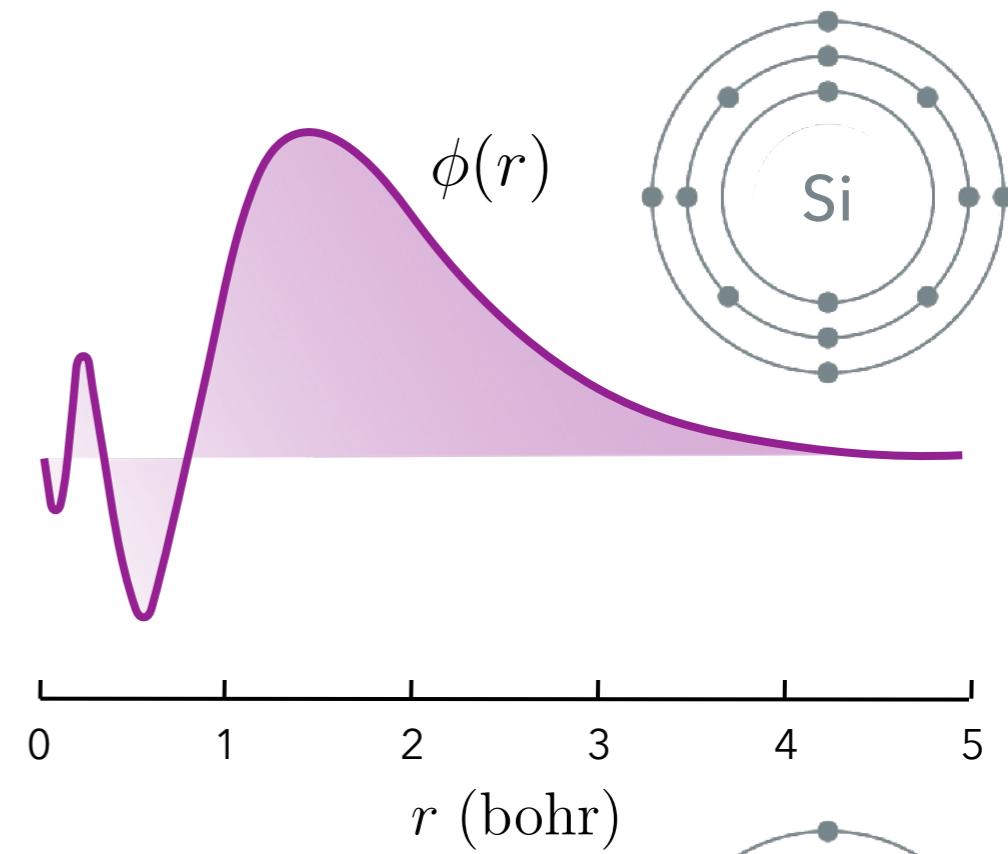
- To manage this infinite sum computationally, only the vectors of magnitude  $|\mathbf{G}|$  below a chosen threshold are included
- This threshold is defined in terms of a **cutoff kinetic energy**

$$\frac{\hbar^2}{2m} |\mathbf{G}|^2 < E_{\text{cutoff}}$$

# Important terminology: pseudopotentials

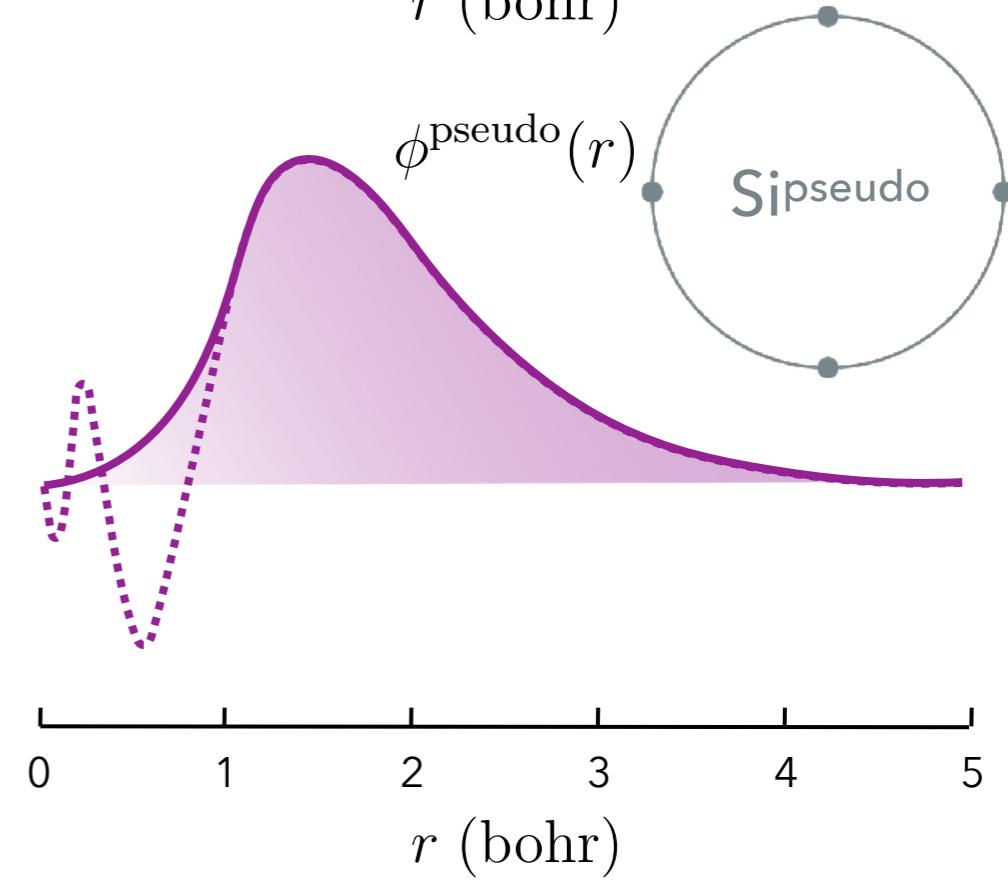
## Strong oscillations

- Near atomic cores, the orbitals exhibit strong oscillations
- Capturing these oscillations require a very high  $E_{\text{cutoff}}$



## Pseudization

- The pseudization consists of cancelling out oscillations by smoothing the nuclear potential
- The resulting pseudopotential does not include the core electrons



# **Supercell approximation**

---