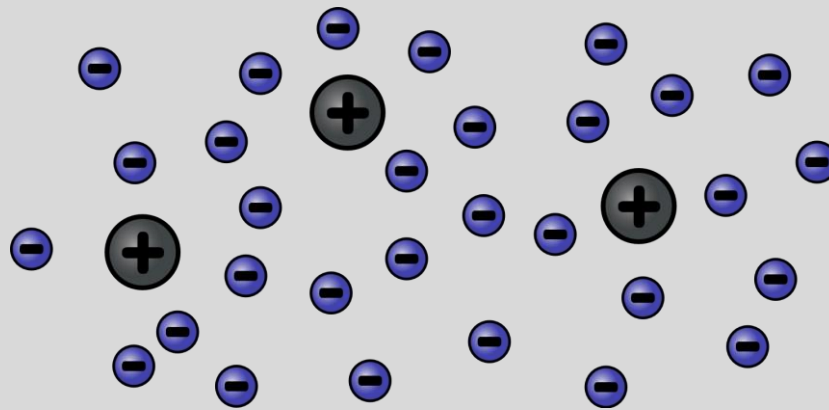
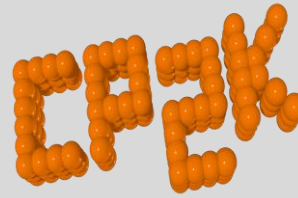


Introduction to



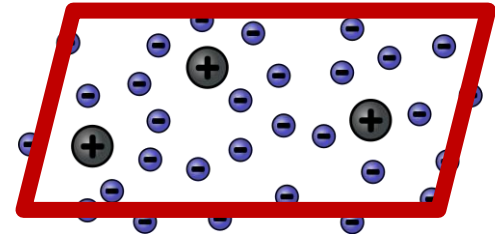
Jan Wilhelm

Regensburg Center for Ultrafast Nanoscopy
University of Regensburg

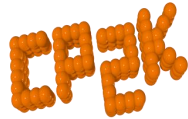
What is a first-principles calculation?

Input:

- atomic coordinates $\{\mathbf{R}_A\}$
- **unit cell** (lattice constants, angles)
- number of electrons N (usually charge = 0)



Software packages:



QUANTUM ESPRESSO



and many more ...

Output:

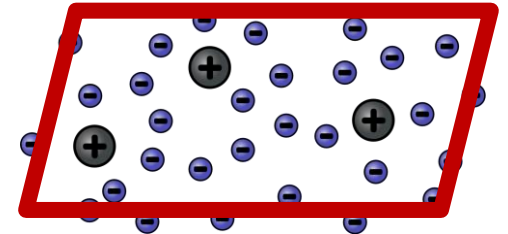
- ground state energy E_0 of electrons and nuclei
- optimized positions of atoms to minimize E_0 (geometry & cell optimization)
- statistical average over atomic positions (molecular dynamics; for liquids)
- electronic band structure of a crystal
- And much more ...

How do I define the atomic coordinates and the cell in CP2K?

Input:

- atomic coordinates $\{\mathbf{R}_A\}$

```
&COORD
  Mo      0.0000    0.0000    9.0635
  S       1.5920    0.9191   10.6271
  S       1.5920    0.9191    7.5000
&END COORD
```

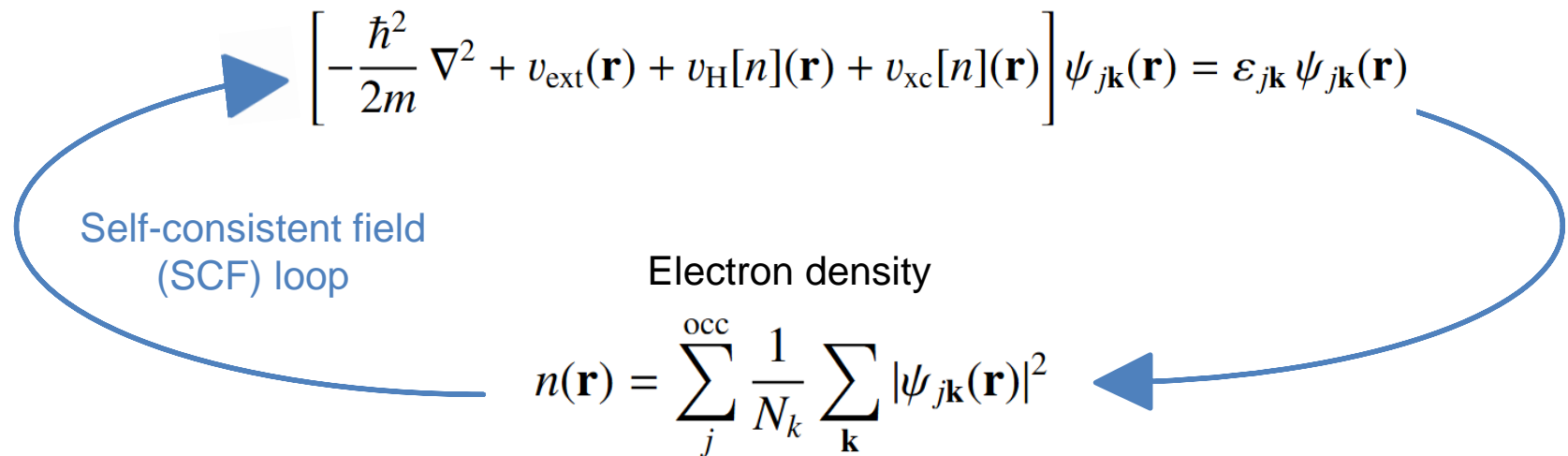


- **unit cell** (lattice constants, angles)

```
&CELL
  ABC              3.184    3.184    18.127
  ALPHA_BETA_GAMMA 90      90      120
  PERIODIC XY
&END CELL
```

How do we compute the ground state energy E_0 ?

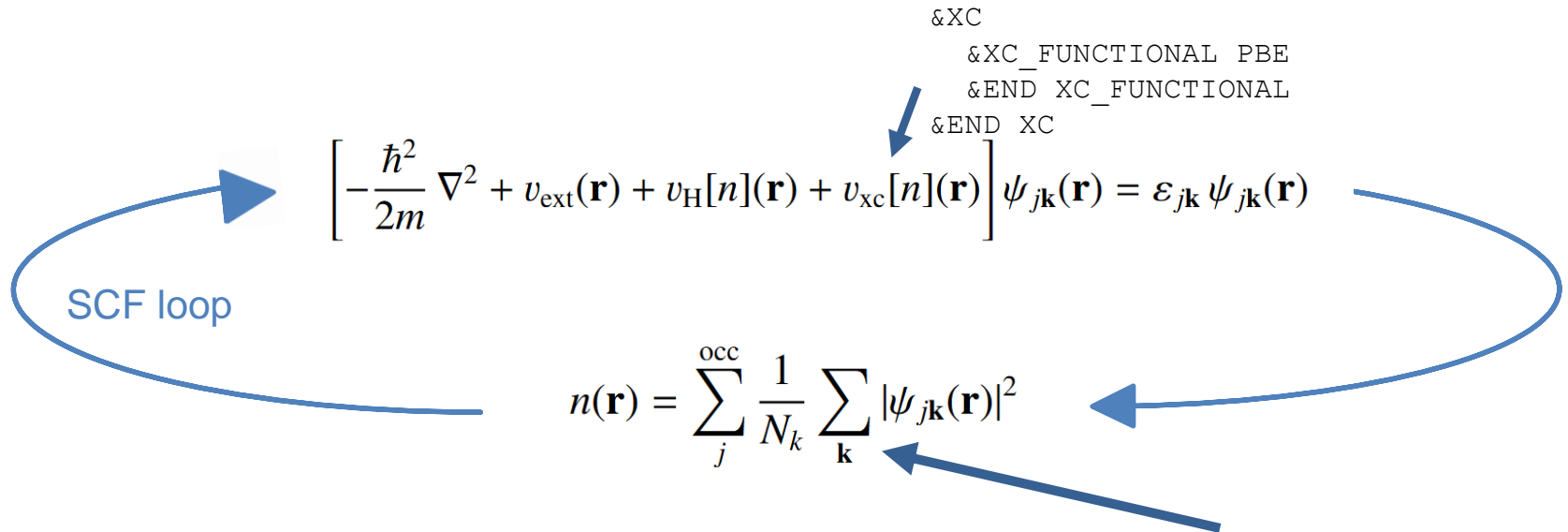
We use Kohn-Sham density functional theory (KS-DFT) with periodic boundary conditions:



Once electron density n is known: compute ground state energy E_0 from total energy functional $E[n]$:

$$E_0 = E[n]$$

Which CP2K input parameters control the KS equations & the SCF?



SCF via „Diagonalization“:

```
&SCF
  SCF_GUESS ATOMIC
  EPS_SCF 1.0E-7
  MAX_SCF 200
  &MIXING
    METHOD BROYDEN_MIXING
    ALPHA 0.1
    BETA 1.5
    NBROYDEN 8
  &END
&END SCF
```

SCF via „Orbital Transformation“
(OT, if SCF convergence problems):

```
&SCF
  SCF_GUESS ATOMIC
  MAX_SCF 50
  &OT
    PRECONDITIONER FULL_SINGLE_INVERSE
    MINIMIZER CG
    LINESEARCH 3PNT
  &END
  EPS_SCF 1.0E-7
  &OUTER_SCF
    MAX_SCF 20
    EPS_SCF 1.0E-7
  &END
&END SCF
```

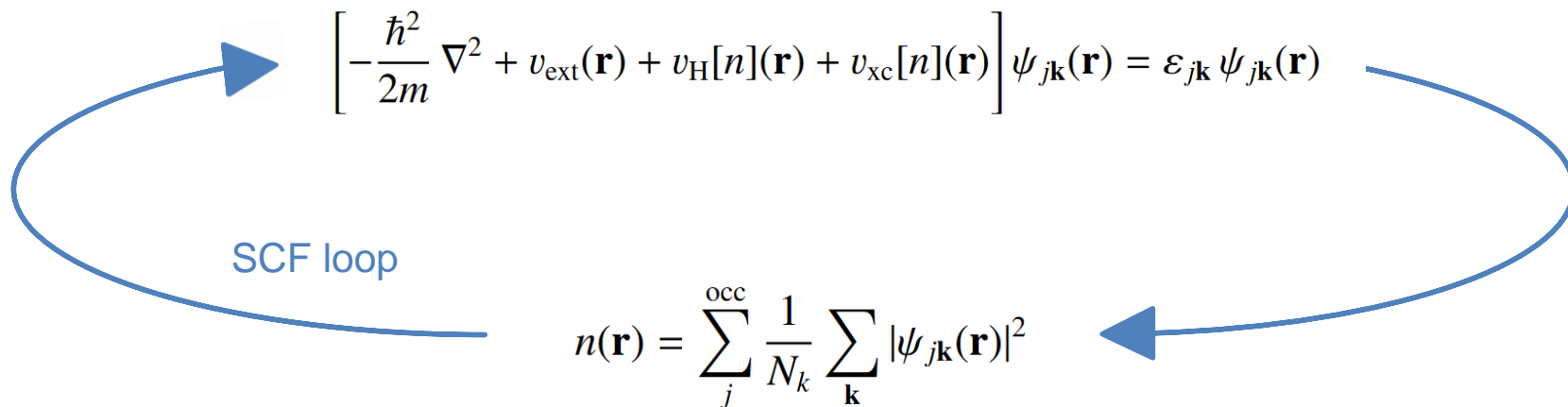
64 k-points (for 2D crystal)

```
&KPOINTS
  SCHEME MONKHORST-PACK 8 8 1
&END KPOINTS
```

For large unit cells (e.g., liquids):

Only use a single k-point (Γ -point)
by not putting &KPOINTS at all

How does the SCF look like in the CP2K output?



CP2K output:

SCF WAVEFUNCTION OPTIMIZATION

Step	Update method	Time	Convergence	Total energy	Change
1	NoMix/Diag. 0.10E+00	0.6	2.12775093	-89.0857269257	-8.91E+01
2	Broy./Diag. 0.10E+00	0.7	0.03335901	-90.7724221181	-1.69E+00

...

56	Broy./Diag. 0.10E+00	0.6	6.8529E-10	-88.7583628545	4.43E-09
----	----------------------	-----	------------	----------------	----------

*** SCF run converged in 56 steps ***

...

Total energy:

$$E_0 = E[n] \quad -88.75836285449478$$

(unit: Hartree)

How can we print the band structure with CP2K?

$$\left[-\frac{\hbar^2}{2m} \nabla^2 + v_{\text{ext}}(\mathbf{r}) + v_{\text{H}}[n](\mathbf{r}) + v_{\text{xc}}[n](\mathbf{r}) \right] \psi_{j\mathbf{k}}(\mathbf{r}) = \varepsilon_{j\mathbf{k}} \psi_{j\mathbf{k}}(\mathbf{r})$$

band structure

Plot band structure along k-path in Brillouin zone:

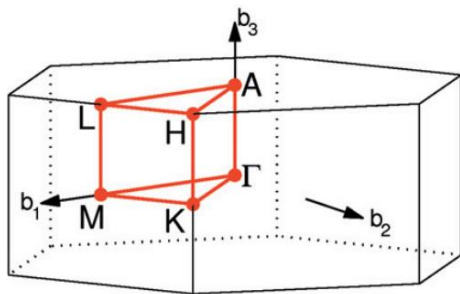


Fig. 13. Brillouin zone of HEX lattice. Path: Γ -M-K- Γ -A-L-H-A|L-M|K-H. An example of band structure using this path is given in Fig. 38.

$\times \mathbf{b}_1$	$\times \mathbf{b}_2$	$\times \mathbf{b}_3$	
0	0	0	Γ
1/3	1/3	0	K
1/2	0	0	M

CP2K input:

```
&PRINT
  &BAND_STRUCTURE
    ADDED_MOS 10
    FILE_NAME bandstructure.bs
  &KPOINT_SET
    NPOINTS 49
    SPECIAL_POINT GAMMA 0.0 0.0 0.0
    SPECIAL_POINT M      0.0 0.5 0.0
    SPECIAL_POINT K      0.33 0.33 0.0
    SPECIAL_POINT GAMMA 0.0 0.0 0.0
  &END KPOINT_SET
&END BAND_STRUCTURE
&END PRINT
```

W. Setyawan, S. Curtarolo, Comput. Mater. Sci. **49**, 299 (2010)

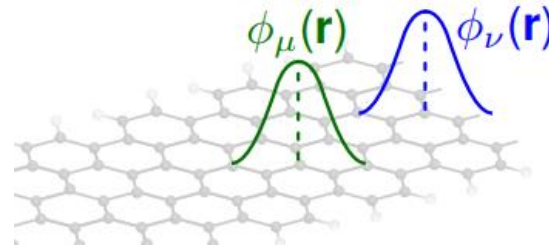
<https://www.sciencedirect.com/science/article/abs/pii/S0927025610002697>

How can we reformulate KS-DFT to get matrix equations? (drop k -index)

Basis set: $\psi_n(\mathbf{r}) = \sum_{\nu=1}^{N_{\text{basis}}} C_{\nu n} \phi_{\nu}(\mathbf{r})$ $\phi_{\nu}(\mathbf{r})$: atom-centered Gaussian basis function

Example: Gaussian-type s-function:

$$\phi_{\nu}(\mathbf{r}) = \left(\frac{\pi}{\alpha}\right)^{3/2} \exp\left(-\alpha|\mathbf{r} - \mathbf{R}_A|^2\right)$$



Also p-, d-, ... functions, typically 30 basis functions per atom: $N_{\text{basis}} \approx 30 N_{\text{atom}}$

$$\left[-\frac{\hbar^2}{2m} \nabla^2 + v_{\text{ext}}(\mathbf{r}) + v_{\text{H}}(\mathbf{r}) + v_{\text{xc}}(\mathbf{r}) \right] \psi_n(\mathbf{r}) = \varepsilon_n \psi_n(\mathbf{r})$$

$$\longrightarrow \int \phi_{\mu}(\mathbf{r}) \left[-\frac{\hbar^2}{2m} \nabla^2 + v_{\text{ext}}(\mathbf{r}) + v_{\text{H}}(\mathbf{r}) + v_{\text{xc}}(\mathbf{r}) \right] \sum_{\nu} C_{\nu n} \phi_{\nu}(\mathbf{r}) d\mathbf{r} = \varepsilon_n \int \phi_{\mu}(\mathbf{r}) \sum_{\nu} C_{\nu n} \phi_{\nu}(\mathbf{r}) d\mathbf{r}$$

$$\longrightarrow \sum_{\nu} h_{\mu\nu} C_{\nu n} = \sum_{\nu} S_{\mu\nu} C_{\nu n} \varepsilon_n \longrightarrow \mathbf{h}\mathbf{C} = \mathbf{S}\mathbf{C}\boldsymbol{\varepsilon}$$

Kohn-Sham equations in Gaussian basis $\phi_{\nu}(\mathbf{r})$ is a matrix equation (input: \mathbf{h} , \mathbf{S} matrix; output: \mathbf{C} , $\boldsymbol{\varepsilon}$)

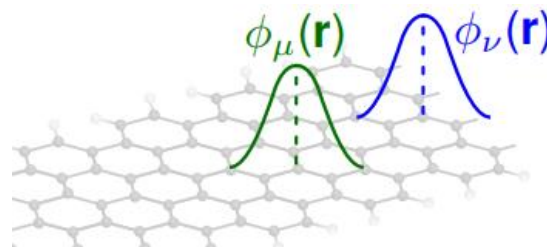
How do we define the Gaussian basis functions in the CP2K input?

Basis set:
$$\psi_n(\mathbf{r}) = \sum_{v=1}^{N_{\text{basis}}} C_{vn} \phi_v(\mathbf{r})$$

$\phi_v(\mathbf{r})$: atom-centered Gaussian basis function

Example: Gaussian-type s-function:

$$\phi_v(\mathbf{r}) = \left(\frac{\pi}{\alpha}\right)^{3/2} \exp\left(-\alpha|\mathbf{r} - \mathbf{R}_A|^2\right)$$



CP2K input:

```
...  
BASIS_SET_FILE_NAME BASIS_MOLOPT  
...  
  
&KIND S  
  BASIS_SET DZVP-MOLOPT-GTH  
  POTENTIAL GTH-PBE  
&END KIND  
  
&KIND Mo  
  BASIS_SET DZVP-MOLOPT-SR-GTH  
  POTENTIAL GTH-PBE  
&END KIND
```

CP2K basis sets:

<https://github.com/cp2k/cp2k/tree/master/data>

for example BASIS_MOLOPT:

```
S SZV-MOLOPT-GTH  
S DZVP-MOLOPT-GTH  
S TZVP-MOLOPT-GTH  
S TZV2P-MOLOPT-GTH  
S TZV2PX-MOLOPT-GTH
```

Increasing size

Increasing completeness



How can we efficiently compute $v_{\mu\nu}^H$?

$$v_{\mu\nu}^H = \int \phi_{\mu}(\mathbf{r}) v_H(\mathbf{r}) \phi_{\nu}(\mathbf{r}) d\mathbf{r}$$

$$v_H(\mathbf{r}) = \int \frac{n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r}'$$

Gaussian and plane-waves scheme (GPW)

Compute electron density on real-space grid points $\{\mathbf{r}_i\}$:

(CP2K source code: `qs_collocate_density.F`)

$$D_{\mu\nu} = \sum_n^{\text{occ}} C_{\mu n} C_{\nu n}$$
$$n(\mathbf{r}_i) = \sum_{\mu\nu} D_{\mu\nu} \phi_{\mu}(\mathbf{r}_i) \phi_{\nu}(\mathbf{r}_i)$$

Fast Fourier Transform (FFT) of electron density:

$$n(\mathbf{G}_k) = \sum_j e^{i\mathbf{G}_k \mathbf{r}_j} n(\mathbf{r}_j)$$

Compute Hartree potential:

$$v_H(\mathbf{G}_k) = \frac{4\pi n(\mathbf{G}_k)}{|\mathbf{G}_k|^2}$$

FFT back to real space:

$$v_H(\mathbf{r}_j) = \sum_k e^{-i\mathbf{G}_k \mathbf{r}_j} v_H(\mathbf{G}_k)$$

A hybrid Gaussian and plane wave density functional scheme

1782

1997

G Lippert, J Hutter, M Parrinello

Molecular Physics 92 (3), 477-487



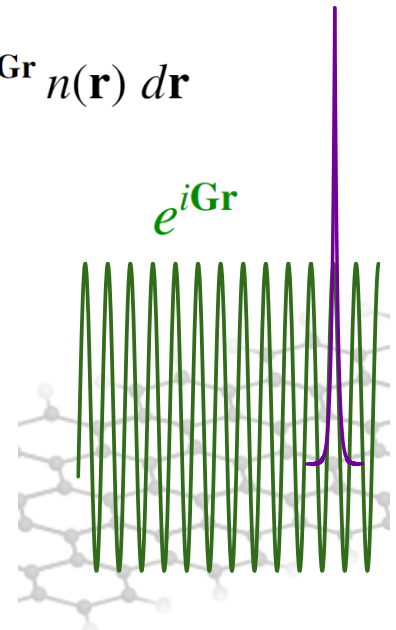
Can we do Fourier transforms of core level functions efficiently? **No.**

FFT $n(\mathbf{G}_k) = \sum_j e^{i\mathbf{G}_k \mathbf{r}_j} n(\mathbf{r}_j)$ is discretization of integral $n(\mathbf{G}) = \int e^{i\mathbf{G}\mathbf{r}} n(\mathbf{r}) d\mathbf{r}$

$n(\mathbf{r}) = \sum_n^{\text{occ}} |\psi_n(\mathbf{r})|^2$ also contains core states; 1s state of H-like atom:

$$\psi_n(\mathbf{r}) \simeq e^{-|\mathbf{r}-\mathbf{R}_A|/r_0}$$

$$r_0 = \frac{a_0}{Z}$$



→ for heavy atoms, Z is large and core levels get extremely localized

→ one needs extremely many plane waves $e^{i\mathbf{G}\mathbf{r}}$ for accurate Fourier trafo of $n(\mathbf{r})$

→ computationally very demanding

→ use pseudopotential (PP) and absorb core electrons into PP: $\frac{Ze^2}{|\mathbf{r} - \mathbf{R}_A|} \rightarrow V_{\text{PP}}^A(\mathbf{r})$



GTH pseudopotentials

Separable dual-space Gaussian pseudopotentials

S Goedecker, M Teter, J Hutter

Physical Review B 54 (3), 1703, 1996

6792

How can we control the Fourier transform and the pseudopotentials in the input?

FFT
$$n(\mathbf{G}_k) = \sum_j e^{i\mathbf{G}_k \cdot \mathbf{r}_j} n(\mathbf{r}_j)$$

Further reading about CUTOFFs:

<https://www.cp2k.org/faq:cutoff>

<https://manual.cp2k.org/trunk/methods/dft/cutoff.html>

`&MGRID
 CUTOFF 600
 REL_CUTOFF 100
&END MGRID`

determines
max $|\mathbf{G}_k|$



CUTOFF 600, REL_CUTOFF 100 are well-converged in most applications

Pseudopotentials

Gaussian and plane-wave scheme

$$\frac{Ze^2}{|\mathbf{r} - \mathbf{R}_A|} \rightarrow V_{\text{PP}}^A(\mathbf{r})$$

```
&KIND S
  BASIS_SET DZVP-MOLOPT-GTH
  POTENTIAL GTH-PBE
&END KIND
```

```
&QS
  METHOD GPW
```

...

```
&END QS
```

Separable dual-space Gaussian pseudopotentials

S Goedecker, M Teter, J Hutter
Physical Review B 54 (3), 1703, 1996

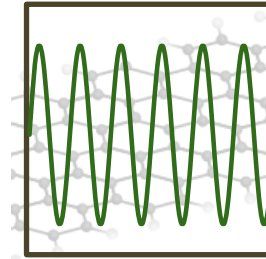
A hybrid Gaussian and plane wave density functional scheme

G Lippert, J Hutter, M Parrinello
Molecular Physics 92 (3), 477-487

Are atom-centered basis functions the only basis set? (not in CP2K)

Plane-wave basis set for KS orbitals:

$$\psi_n(\mathbf{r}) = \sum_{\mathbf{G} \text{ with } |\mathbf{G}| < G_c} C_{\mathbf{G}n} e^{i\mathbf{G} \cdot \mathbf{r}}$$



Use all plane waves that are periodic w.r.t. simulation box and are below cutoff G_c

Advantage of plane waves over Gaussian basis functions:

- systematic convergence of basis set by increasing G_c
- no pretabulation of basis sets necessary

Disadvantages of plane waves compared to Gaussian basis functions:

- inefficient for all systems with large vacuum region, for example single molecule, 2D materials, surface, metal-organic framework, some liquids
- Gaussian basis is usually smaller than plane-wave basis
→ Usually Gaussian-based codes can deal with more atoms in the unit cell

How can we optimize / move the atomic positions and the cell?

Standard KS-DFT calculation:

```
&GLOBAL  
  ...  
  RUN_TYPE ENERGY  
&END GLOBAL
```

Geometry optimization (optimize atomic positions, keep cell fixed):

```
&GLOBAL  
  ...  
  RUN_TYPE GEO_OPT  
&END GLOBAL
```

Cell optimization (optimize atomic positions and unit cell):

```
&GLOBAL  
  ...  
  RUN_TYPE CELL_OPT  
&END GLOBAL
```

Molecular dynamics (move atoms according to temperature):

```
&GLOBAL  
  ...  
  RUN_TYPE MD  
&END GLOBAL
```

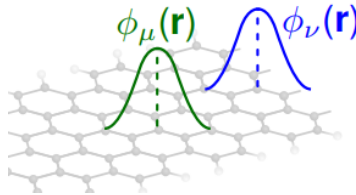
The full CP2K input

```
&GLOBAL
  PROJECT   MoS2
  RUN_TYPE  ENERGY
&END GLOBAL
&FORCE_EVAL
  &DFT
    BASIS_SET_FILE_NAME  BASIS_MOLOPT
    POTENTIAL_FILE_NAME  GTH_POTENTIALS
  &MGRID
    CUTOFF    500
    REL_CUTOFF 100
  &END MGRID
  &QS
    METHOD GPW
    EPS_DEFAULT 1.0E-12
    EPS_PGF_ORB 1.0E-12
  &END QS
  &SCF
    SCF_GUESS ATOMIC
    EPS_SCF 1.0E-9
    MAX_SCF 500
    &MIXING
      METHOD BROYDEN_MIXING
      ALPHA 0.1
      BETA 1.5
      NBROYDEN 8
    &END
  &END SCF
  &XC
    &XC_FUNCTIONAL PBE
    &END XC_FUNCTIONAL
  &END XC
  &KPOINTS
    SCHEME MONKHORST-PACK 8 8 1
  &END KPOINTS
```

...

```
&PRINT
  &BAND_STRUCTURE
    ADDED_MOS 10
    FILE_NAME bandstructure.bs
  &KPOINT_SET
    NPOINTS 49
    SPECIAL_POINT GAMMA 0.0 0.0 0.0
    SPECIAL_POINT K      0.333333 0.333333 0.0
    SPECIAL_POINT M      0.0 0.5 0.0
    SPECIAL_POINT GAMMA 0.0 0.0 0.0
  &END KPOINT_SET
  &END BAND_STRUCTURE
&END PRINT
&END DFT
&SUBSYS
  &CELL
    ABC          3.184 3.184 18.127
    ALPHA_BETA_GAMMA 90 90 120
    PERIODIC XY
  &END CELL
  &KIND S
    BASIS_SET DZVP-MOLOPT-GTH
    POTENTIAL GTH-PBE
  &END KIND
  &KIND Mo
    BASIS_SET DZVP-MOLOPT-SR-GTH
    POTENTIAL GTH-PBE
  &END KIND
  &COORD
Mo      0.00000000      0.00000000      9.06355632
S       1.59203323      0.91916082     10.62711265
S       1.59203323      0.91916082      7.50000000
  &END COORD
  &TOPOLOGY
    &CENTER_COORDINATES
    &END
  &END
&END SUBSYS
&END FORCE_EVAL
```

Summary of KS-DFT algorithm in CP2K

- Expand Kohn-Sham orbitals in Gaussian basis: $\psi_n(\mathbf{r}) = \sum_{v=1}^{N_{\text{basis}}} C_{vn} \phi_v(\mathbf{r})$
- 
- Iterate Kohn-Sham equations; for crystals with \mathbf{k} -index:

SCF loop

$$\left[-\frac{\hbar^2}{2m} \nabla^2 + v_{\text{ext}}(\mathbf{r}) + v_{\text{H}}[n](\mathbf{r}) + v_{\text{xc}}[n](\mathbf{r}) \right] \psi_{j\mathbf{k}}(\mathbf{r}) = \varepsilon_{j\mathbf{k}} \psi_{j\mathbf{k}}(\mathbf{r})$$

$$n(\mathbf{r}) = \sum_j^{\text{occ}} \frac{1}{N_k} \sum_{\mathbf{k}} |\psi_{j\mathbf{k}}(\mathbf{r})|^2$$

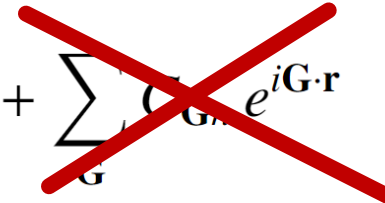
- Compute electron density $n(\mathbf{r})$ from $\psi_n(\mathbf{r})$ on real-space grid \mathbf{r}_j and Fourier transform $n(\mathbf{r})$ to compute Hartree potential (Gaussian and plane waves scheme)

$$n(\mathbf{G}_k) = \sum_j e^{i\mathbf{G}_k \mathbf{r}_j} n(\mathbf{r}_j) \quad v_{\text{H}}(\mathbf{G}_k) = \frac{4\pi n(\mathbf{G}_k)}{|\mathbf{G}_k|^2} \quad v_{\text{H}}(\mathbf{r}_j) = \sum_k e^{-i\mathbf{G}_k \mathbf{r}_j} v_{\text{H}}(\mathbf{G}_k)$$

Question to you:

Why is the Gaussian and plane-wave scheme (GPW) in CP2K named this way?

(a) Because KS orbitals are expanded in a basis of Gaussians and plane waves:

$$\psi_n(\mathbf{r}) = \sum_{\mu} C_{\mu n} \phi_{\mu}(\mathbf{r}) + \sum_{\mathbf{G}} C_{\mathbf{G}n} e^{i\mathbf{G}\cdot\mathbf{r}}$$


(b) Because KS orbitals are expanded in Gaussians:

$$\psi_n(\mathbf{r}) = \sum_{\mu} C_{\mu n} \phi_{\mu}(\mathbf{r})$$

whereas plane waves are used for the Fourier transformation of the electron density n :

$$n(\mathbf{G}_k) = \sum_j e^{i\mathbf{G}_k \cdot \mathbf{r}_j} n(\mathbf{r}_j)$$

Correct answer: (b).

The KS orbitals are only expanded in Gaussians, not in plane waves!