# MODULE 2:
# QUANTUM MECHANICS

## Practice: Quantum ESPRESSO

# I. What is Quantum ESPRESSO?

# DFT software

| | | |
|---|---|---|
| QUANTUM ESPRESSO | PW-DFT, PP, US-PP, PAW<br>http://www.quantum-espresso.org | FREE |
| abinit.org | PW-DFT, PP, PAW<br>http://www.abinit.org | FREE |
| μCPMD | DFT PW, PP, Car-Parrinello<br>http://www.cpmd.org | FREE |
| VASP | DFT PP, US-PP, PAW<br>[moderate accuracy, fast]<br>http://www.vasp.at | $3000 |
| WIEN2k | DFT full-potential linearized augmented plane-wave (FLAPW) [accurate, slow]<br>http://www.wien2k.at | $500 |
| | Hartree-Fock, higher order correlated electron approaches<br>http://www.gaussian.com | $3000 |

# Quantum ESPRESSO

# Quantum ESPRESSO

- Quantum ESPRESSO is an integrated suite of Open-Source computer codes for electronic-structure calculations and materials modeling at the nanoscale.

- It is based on density-functional theory, plane waves, and pseudopotentials.

- Core set of codes, plugins for more advanced tasks and third party packages
- Open initiative coordinated by the Quantum ESPRESSO Foundation, across Italy.
- Contributed to by developers across the world
- Regular hands-on workshops in Trieste, Italy
- Open-source code: **FREE** (unlike VASP...)

# Performance

- Small jobs (a few atoms) can be run on single node
  - Includes determining convergence parameters, lattice constants
  - Can use OpenMP parallelization on multicore machines

- Large jobs (~10's to ~100's atoms) can run in parallel using MPI to 1000's of cores
  - Includes molecular dynamics, large geometry relaxation, phonons

- Parallel performance tied to BLAS/LAPACK (linear algebra routines) and 3D FFT (fast Fourier transform)

- New GPU-enabled version available

# Usability

- Documented online:

  www.quantum-espresso.org/users-manual/

- Multiple mailing lists for questions (with archives)

  www.quantum-espresso.org/forum/

- Many tutorials and guides available

  www.quantum-espresso.org/tutorials/

- Visualization tools and analysis tools available

# Availability

- Free download of source code from

  www.quantum-espresso.org/download/

  http://qe-forge.org/gf/project/q-e/frs/?action=FrsReleaseBrowse&frs_package_id=18

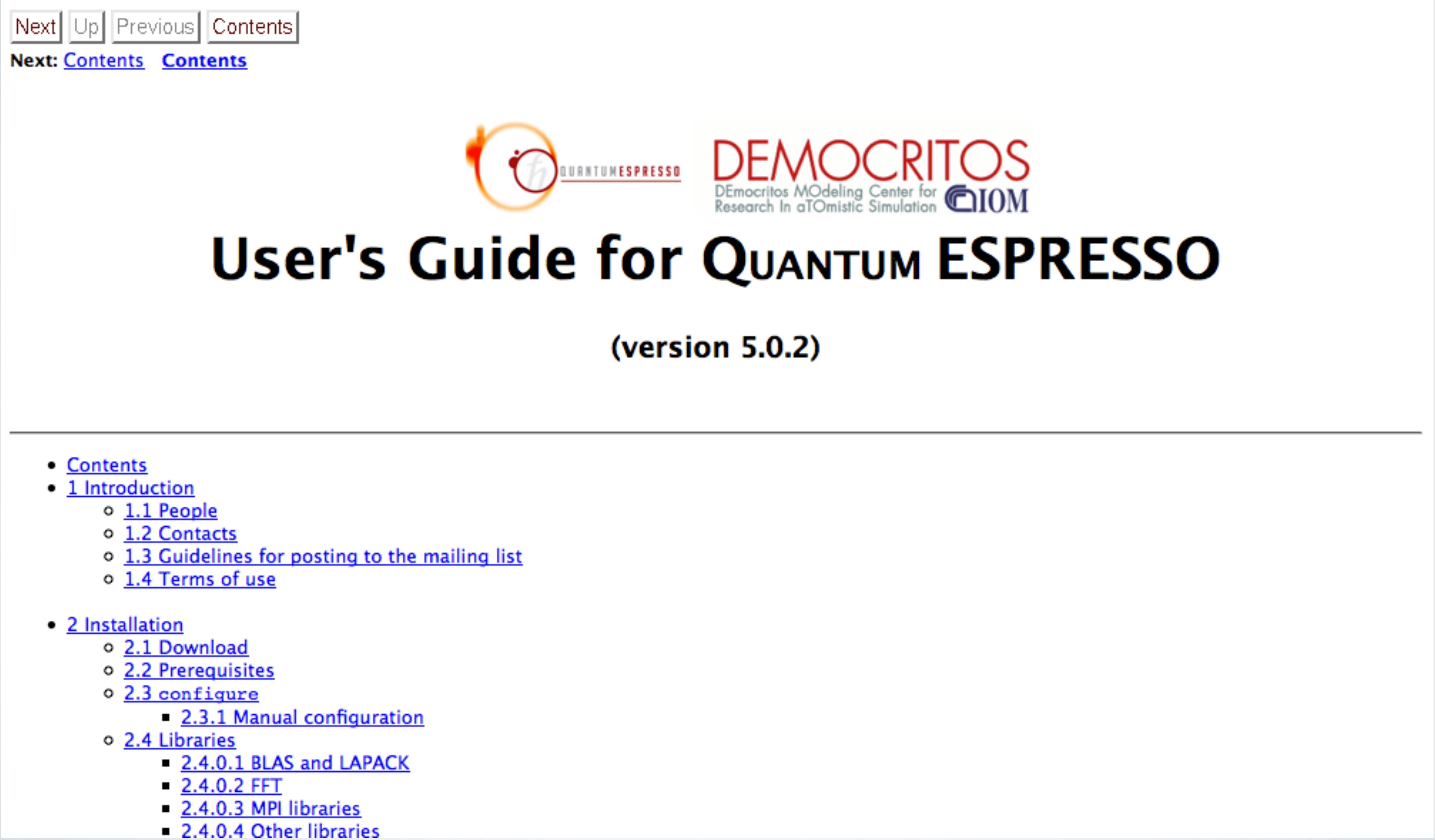
- Some binaries available via Debian

- Supported by Linux and Mac OS X, Windows via Cygwin

- Requirements:
  - Fortran 90/95 and C compilers + standard libraries
  - MPI (for parallel runs)
  - make
  - LAPACK/BLAS for linear algebra; SCALAPACK optional
  - optimized FFT libraries (optional)

# Installation

www.quantum-espresso.org/wp-content/uploads/Doc/user_guide/



**Next:** Contents **Contents**

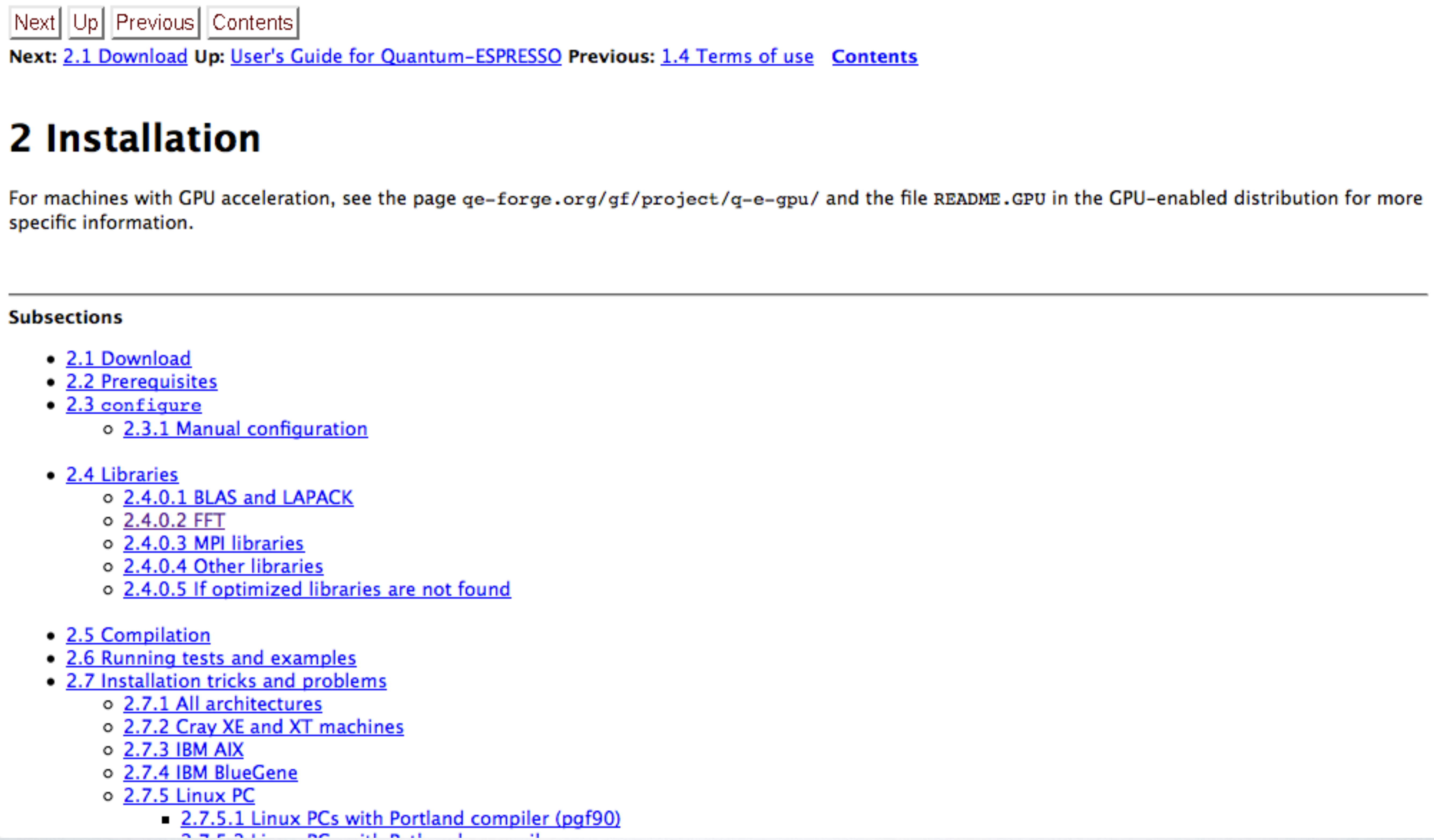
## User's Guide for Quantum ESPRESSO

(version 5.0.2)

---

# Installation

www.quantum-espresso.org/wp-content/uploads/Doc/user_guide/node7.html



**Next:** 2.1 Download **Up:** User's Guide for Quantum-ESPRESSO **Previous:** 1.4 Terms of use **Contents**

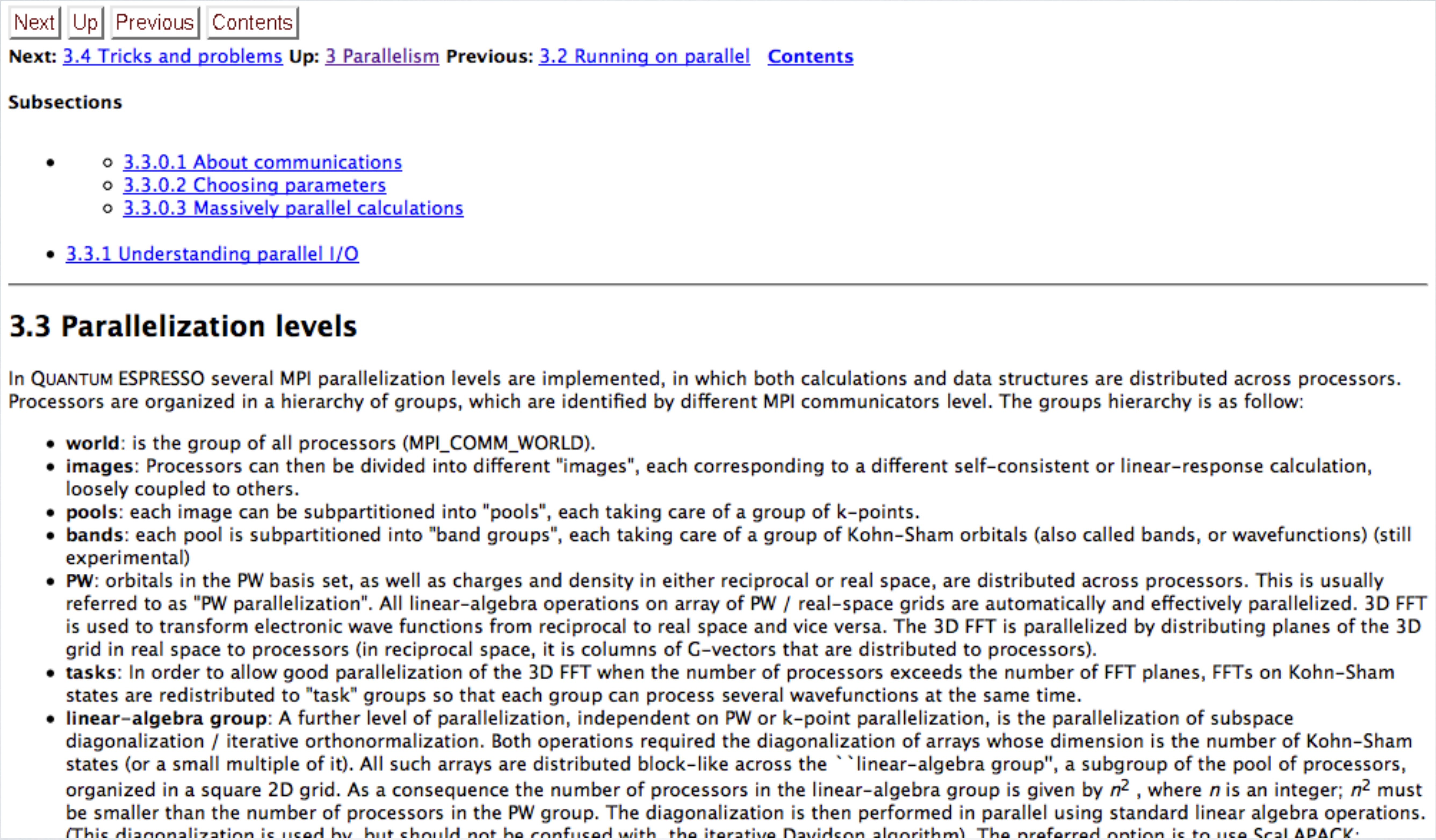
## 2 Installation

For machines with GPU acceleration, see the page `qe-forge.org/gf/project/q-e-gpu/` and the file `README.GPU` in the GPU-enabled distribution for more specific information.

---

**Subsections**

# Parallel performance



**Next:** 3.4 Tricks and problems **Up:** 3 Parallelism **Previous:** 3.2 Running on parallel **Contents**

**Subsections**

- 
  - 3.3.0.1 About communications
  - 3.3.0.2 Choosing parameters
  - 3.3.0.3 Massively parallel calculations
- 3.3.1 Understanding parallel I/O

---

## 3.3 Parallelization levels

In QUANTUM ESPRESSO several MPI parallelization levels are implemented, in which both calculations and data structures are distributed across processors. Processors are organized in a hierarchy of groups, which are identified by different MPI communicators level. The groups hierarchy is as follow:

- **world**: is the group of all processors (MPI_COMM_WORLD).
- **images**: Processors can then be divided into different "images", each corresponding to a different self-consistent or linear-response calculation, loosely coupled to others.
- **pools**: each image can be subpartitioned into "pools", each taking care of a group of k-points.
- **bands**: each pool is subpartitioned into "band groups", each taking care of a group of Kohn-Sham orbitals (also called bands, or wavefunctions) (still experimental)
- **PW**: orbitals in the PW basis set, as well as charges and density in either reciprocal or real space, are distributed across processors. This is usually referred to as "PW parallelization". All linear-algebra operations on array of PW / real-space grids are automatically and effectively parallelized. 3D FFT is used to transform electronic wave functions from reciprocal to real space and vice versa. The 3D FFT is parallelized by distributing planes of the 3D grid in real space to processors (in reciprocal space, it is columns of G-vectors that are distributed to processors).
- **tasks**: In order to allow good parallelization of the 3D FFT when the number of processors exceeds the number of FFT planes, FFTs on Kohn-Sham states are redistributed to "task" groups so that each group can process several wavefunctions at the same time.
- **linear-algebra group**: A further level of parallelization, independent on PW or k-point parallelization, is the parallelization of subspace diagonalization / iterative orthonormalization. Both operations required the diagonalization of arrays whose dimension is the number of Kohn-Sham states (or a small multiple of it). All such arrays are distributed block-like across the ``linear-algebra group'', a subgroup of the pool of processors, organized in a square 2D grid. As a consequence the number of processors in the linear-algebra group is given by $n^2$ , where $n$ is an integer; $n^2$ must be smaller than the number of processors in the PW group. The diagonalization is then performed in parallel using standard linear algebra operations. (This diagonalization is used by, but should not be confused with, the iterative Davidson algorithm). The preferred option is to use ScaLAPACK;

# What can Quantum ESPRESSO do?

- Ground-state calculations
- Structural optimization
- Transition states and minimum energy paths
- Ab-initio molecular dynamics
- Response properties (DFPT)
- Spectroscopic properties
- Quantum transport

# What can Quantum ESPRESSO do?

- **Ground-state calculations**
  - Self-consistent total energies, forces, stresses;
  - Kohn-Sham orbitals;
  - Separable norm-conserving and ultrasoft (Vanderbilt) pseudo-potentials, PAW (Projector Augmented Waves);
  - Several exchange-correlation functionals: from LDA to generalized-gradient corrections (PW91, PBE, B88-P86, BLYP) to meta-GGA, exact exchange (HF) and hybrid functionals (PBE0, B3LYP, HSE);
  - Van der Waals corrections (DFT-D) or nonlocal VdW functionals (vdw-DF);
  - Hubbard U for conducting/insulating transition (DFT+U).

# What can Quantum ESPRESSO do?

- **Structural optimization**
  - GDIIS with quasi-Newton BFGS preconditioning;
  - Damped dynamics.

- **Transition states and minimum energy paths**
  - Nudged Elastic Band method (PWneb);
  - Meta-Dynamics using the PLUMED plug-in.

- **Ab-initio molecular dynamics**
  - Car-Parrinello Molecular Dynamics (CP);
  - Born-Oppenheimer Molecular Dynamics (PWscf).

# What can Quantum ESPRESSO do?

- **Spectroscopic properties**
  - K- and $L_I$-edge X-ray Absorption Spectra (XSpectra);
  - Time-Dependent Density Functional Perturbation Theory (TDDFPT);
  - Electronic excitations with Many-Body Perturbation Theory using YAMBO package.
  - Electronic excitations with Many-Body Perturbation Theory using GWL package.
- **Quantum transport**
  - Ballistic Transport using PWCOND module;
  - Coherent Transport from Maximally Localized Wannier Functions using WanT code;
  - Maximally-localized Wannier functions and transport properties using WANNIER90 code.

# What can Quantum ESPRESSO do?

- **Platforms**
  - Runs on almost every conceivable current architecture
  - Large parallel machines (IBM BlueGene, Cray XT)
  - Workstations running Linux, Windows, Mac OS-X
  - Clusters with various connectivity (gigE, infiniband)
  - GPU via the QE-GPU package;
  - Playstations!
  - Cellphones!

# II. Running Quantum ESPRESSO

# Overview

- Performing a calculation in DFT requires:
  - Determining where atoms are located, and what quantities will be calculated (like all simulations)
- But there's a lot of control over the *approximations* used:
  - How electrons in atomic cores will be treated (pseudopotentials, PAW, etc.)
  - How exchange-correlation interaction will be approximated (LDA, GGA, or more)
  - How many basis functions (planewaves) are used
  - How k-point integration is done

# Overview

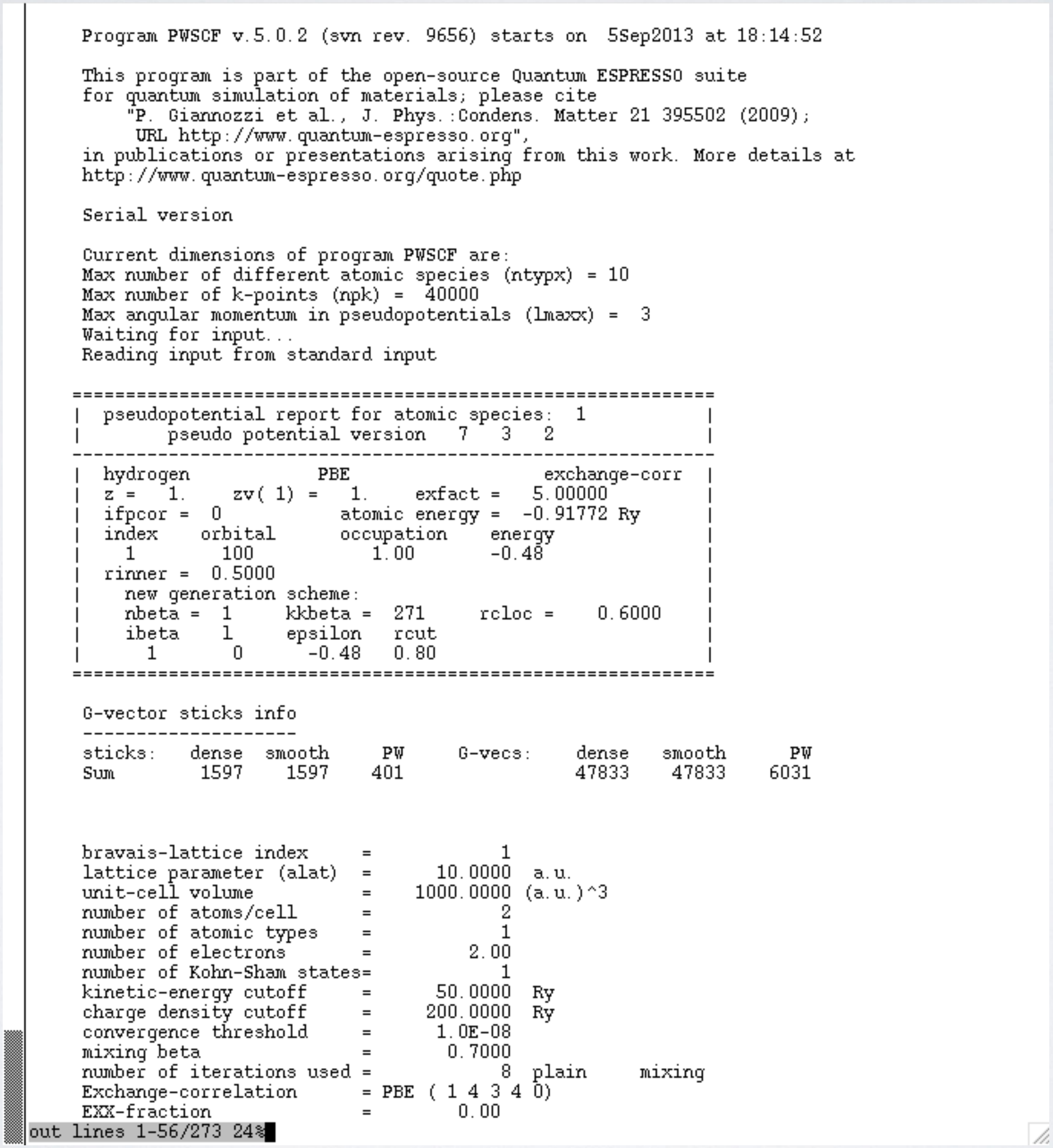
- **All* DFT calculations begin with convergence analysis and error estimation.**

  *That are worth anything.

  - Verification: are you computing the right mathematics?
  - Validation: does your calc. match (experimental) reality?

# Command line interface

- Quantum ESPRESSO is command-line based



```
     Program PWSCF v.5.0.2 (svn rev. 9656) starts on  5Sep2013 at 18:14:52

     This program is part of the open-source Quantum ESPRESSO suite
     for quantum simulation of materials; please cite
         "P. Giannozzi et al., J. Phys.:Condens. Matter 21 395502 (2009);
          URL http://www.quantum-espresso.org",
     in publications or presentations arising from this work. More details at
     http://www.quantum-espresso.org/quote.php

     Serial version

     Current dimensions of program PWSCF are:
     Max number of different atomic species (ntypx) = 10
     Max number of k-points (npk) =  40000
     Max angular momentum in pseudopotentials (lmaxx) =  3
     Waiting for input...
     Reading input from standard input

    =============================================================
    |  pseudopotential report for atomic species:  1            |
    |        pseudo potential version   7   3   2               |
    -------------------------------------------------------------
    |  hydrogen            PBE                  exchange-corr   |
    |  z =   1.    zv( 1) =   1.    exfact =   5.00000          |
    |  ifpcor =  0           atomic energy =  -0.91772 Ry       |
    |  index    orbital      occupation    energy               |
    |    1        100           1.00       -0.48                |
    |  rinner =  0.5000                                         |
    |    new generation scheme:                                 |
    |    nbeta =  1     kkbeta =  271     rcloc =    0.6000     |
    |    ibeta    1     epsilon   rcut                          |
    |      1       0      -0.48   0.80                          |
    =============================================================

     G-vector sticks info
     --------------------
     sticks:   dense  smooth     PW     G-vecs:    dense   smooth      PW
     Sum        1597    1597    401                47833    47833    6031



     bravais-lattice index     =            1
     lattice parameter (alat)  =      10.0000  a.u.
     unit-cell volume          =    1000.0000 (a.u.)^3
     number of atoms/cell      =            2
     number of atomic types    =            1
     number of electrons       =         2.00
     number of Kohn-Sham states=            1
     kinetic-energy cutoff     =      50.0000  Ry
     charge density cutoff     =     200.0000  Ry
     convergence threshold     =      1.0E-08
     mixing beta               =       0.7000
     number of iterations used =            8  plain     mixing
     Exchange-correlation      = PBE ( 1 4 3 4 0)
     EXX-fraction              =        0.00
out lines 1-56/273 24%
```

# Files and programs

- Quantum ESPRESSO works with an input script describing the calculation and pseudopotential files, and produces

| | |
|---|---|
| `stdout` | written to the screen (unless redirected) summary output from run |
| `.wfc` | wavefunction (binary format) |
| `.save/` | directory containing detailed output for visualization / analysis |
| `charge-density.dat` | charge density (binary) |
| `data-file.xml` | detailed XML formated output |
| `_US.van` | pseudopotential files (naming depends on type) |
| `K00001/eigenval.xml` | eigenvalues at each k-point |

# Files and programs

- General type of execution using `pw.x` involves
  - `pw.x < script.inp > script.out`
  - Input script determines directories where output is written, in addition to everything about how job runs
  - Redirection of stdout saves output for analysis

# III. Convergence criteria

# 1. Exchange-Correlation Functionals

**Local density approximation (LDA)**

- Based on Ceperley & Alder's calculations for the uniform electron gas by quantum Monte Carlo (a stochastic method for quantum particles)

**Generalized gradient approximations (GGA: PW91, PBE)**

- Gradients of the density are introduced
- Preserve analytic scaling features of the unknown exact functional

**Meta-GGA (TPSS)**

- Include information about curvature of the density

**Hybrid density functionals (B3LYP, HSE)**

- Based on GGA or meta-GGA approximations
- Add some non-local Hartree-Fock exchange to the functional

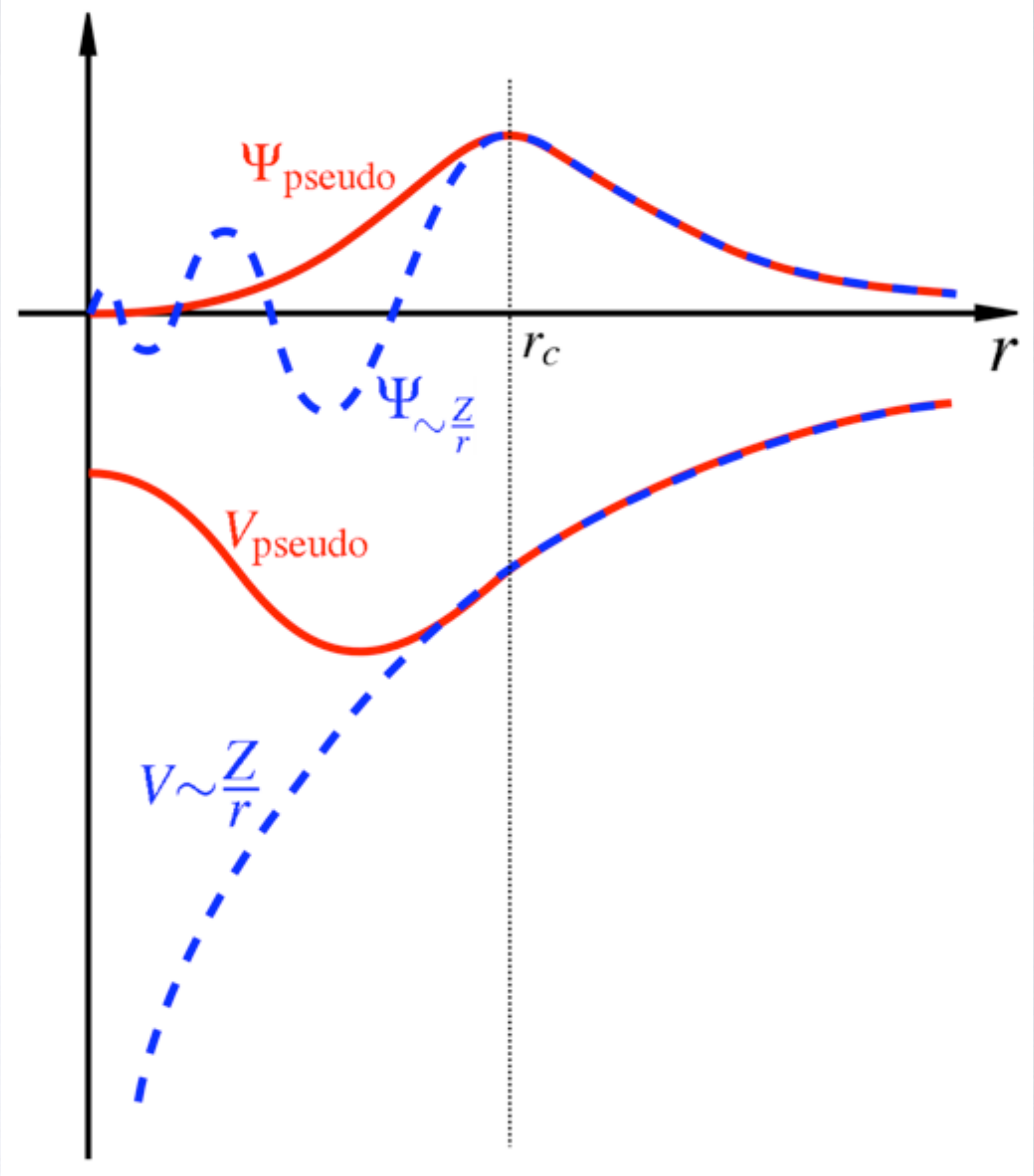
# 2. Pseudopotentials

## The pseudopotential and the wave function

- Real potential and wave function are shown in blue

- Pseudopotential and pseudo wave function in red

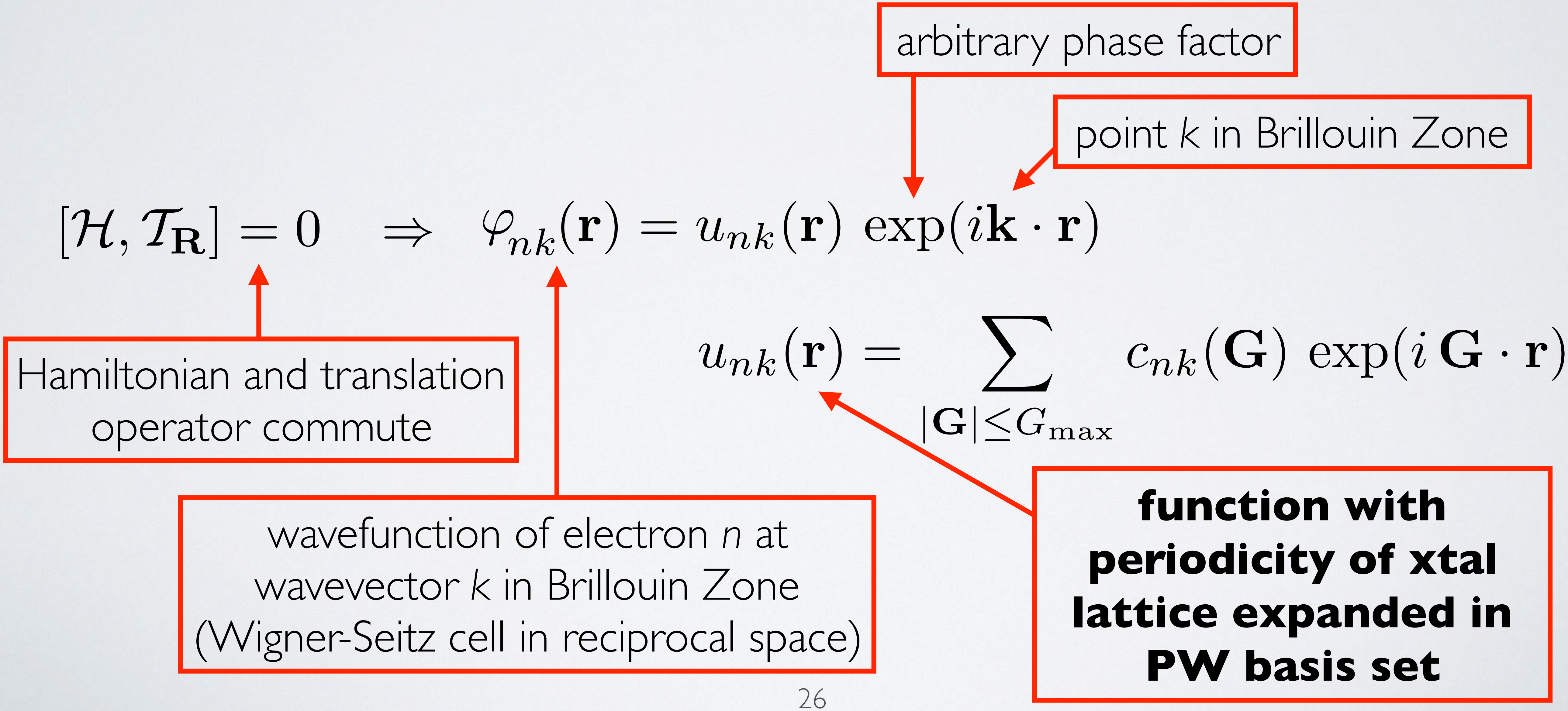
- Outside the cutoff region (vertical black line) the two are identical



# 3. Basis Set Choice

## Basis set choices

- For molecules: use atomic orbitals, or localized functions like Gaussians
- For solids, periodic functions such as sines and cosines (i.e., plane waves)
- Use **Bloch Theorem** for periodic solids:



arbitrary phase factor

point *k* in Brillouin Zone

$$[\mathcal{H}, \mathcal{T}_{\mathbf{R}}] = 0 \quad \Rightarrow \quad \varphi_{nk}(\mathbf{r}) = u_{nk}(\mathbf{r})\ \exp(i\mathbf{k}\cdot\mathbf{r})$$

$$u_{nk}(\mathbf{r}) = \sum_{|\mathbf{G}|\le G_{\max}} c_{nk}(\mathbf{G})\ \exp(i\,\mathbf{G}\cdot\mathbf{r})$$

Hamiltonian and translation operator commute

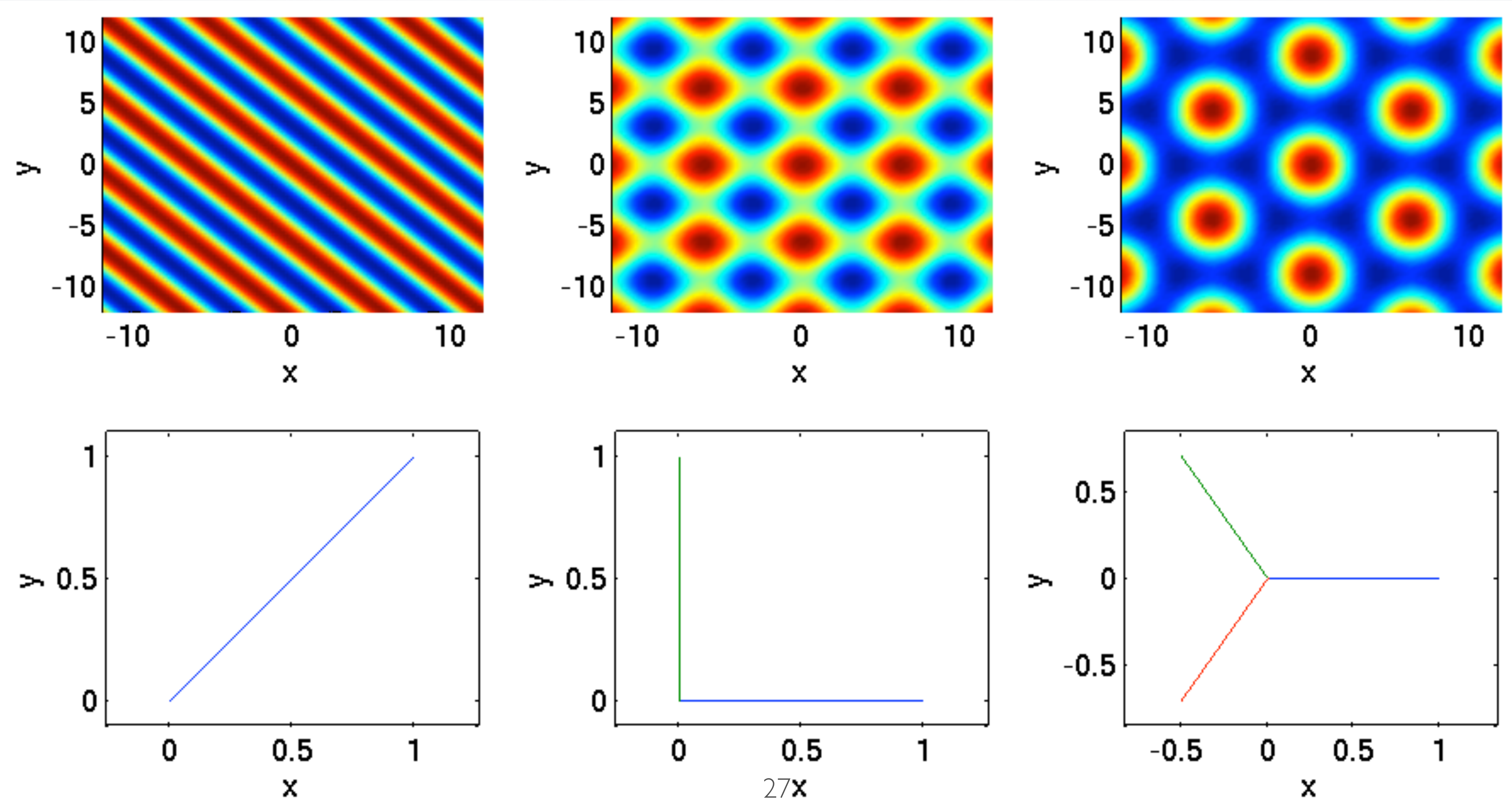
wavefunction of electron *n* at wavevector *k* in Brillouin Zone (Wigner-Seitz cell in reciprocal space)

**function with periodicity of xtal lattice expanded in PW basis set**

# 4. Basis Set Convergence: Plane Wave Basis Set

- Superposition of plane waves to represent orbitals

$$u_{nk}(\mathbf{r}) = \sum_{|\mathbf{G}|\leq G_{\max}} c_{nk}(\mathbf{G})\, \exp(i\,\mathbf{G}\cdot\mathbf{r})$$



# Reciprocal lattice vectors, G

- Superposition of plane waves to represent orbitals

$$u_{nk}(\mathbf{r}) = \sum_{|\mathbf{G}| \leq G_{\max}} c_{nk}(\mathbf{G}) \exp(i\,\mathbf{G} \cdot \mathbf{r})$$

A particular plane wave: $f_i(\mathbf{r}) = c_{i,\mathbf{k}+\mathbf{G}} e^{-i(\mathbf{k}+\mathbf{G}) \cdot \mathbf{r}}$

Its kinetic energy: $E_{kin} = f_i(\mathbf{r}) \left[ -\frac{\hbar^2}{2m} \nabla^2 \right] f_i(\mathbf{r}) = \frac{\hbar^2}{2m} |\mathbf{k} + \mathbf{G}|^2$

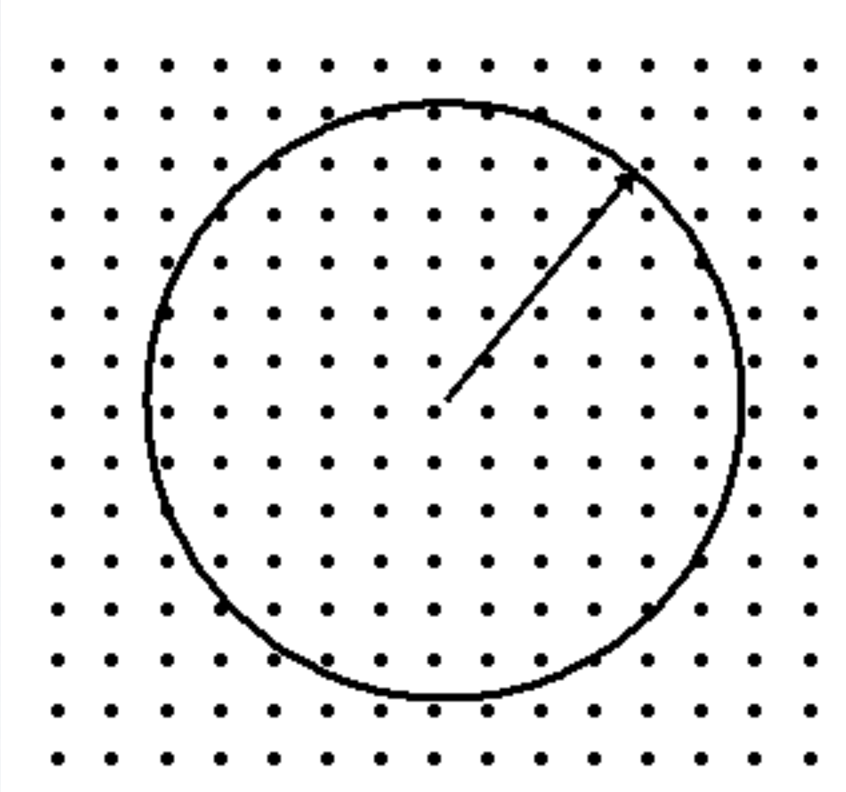
- Low KE plane waves typically contribute more than higher KE
  => we can truncate the approximation to $u_{nk}(r)$ at $E_{cut}$

http://voh.chem.ucla.edu/vohtar/winter05/classes/115B/pdf/DFT_notes_Cocula_Carter.pdf

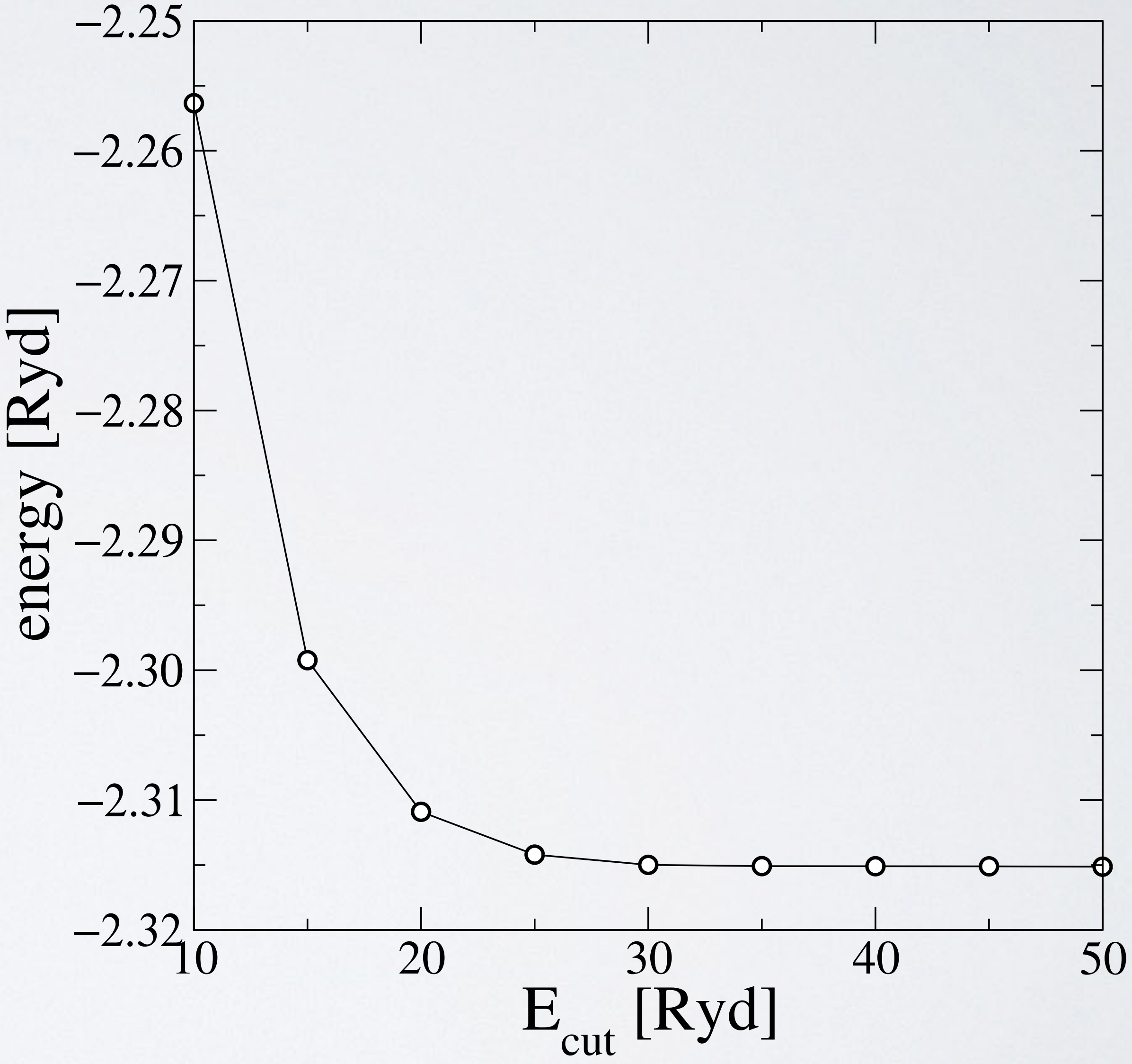
# Reciprocal lattice vectors, G

- Superposition of plane waves to represent orbitals

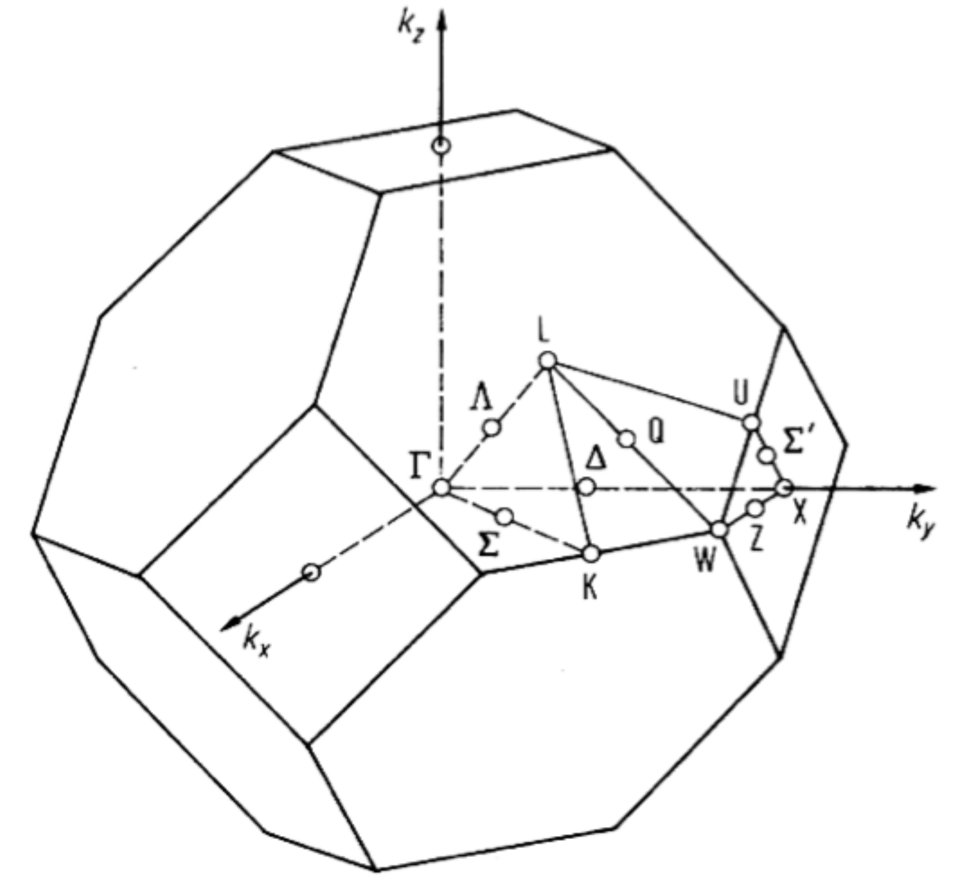
$$u_{nk}(\mathbf{r}) = \sum_{|\mathbf{G}|\leq G_{\max}} c_{nk}(\mathbf{G}) \exp(i\,\mathbf{G}\cdot\mathbf{r})$$

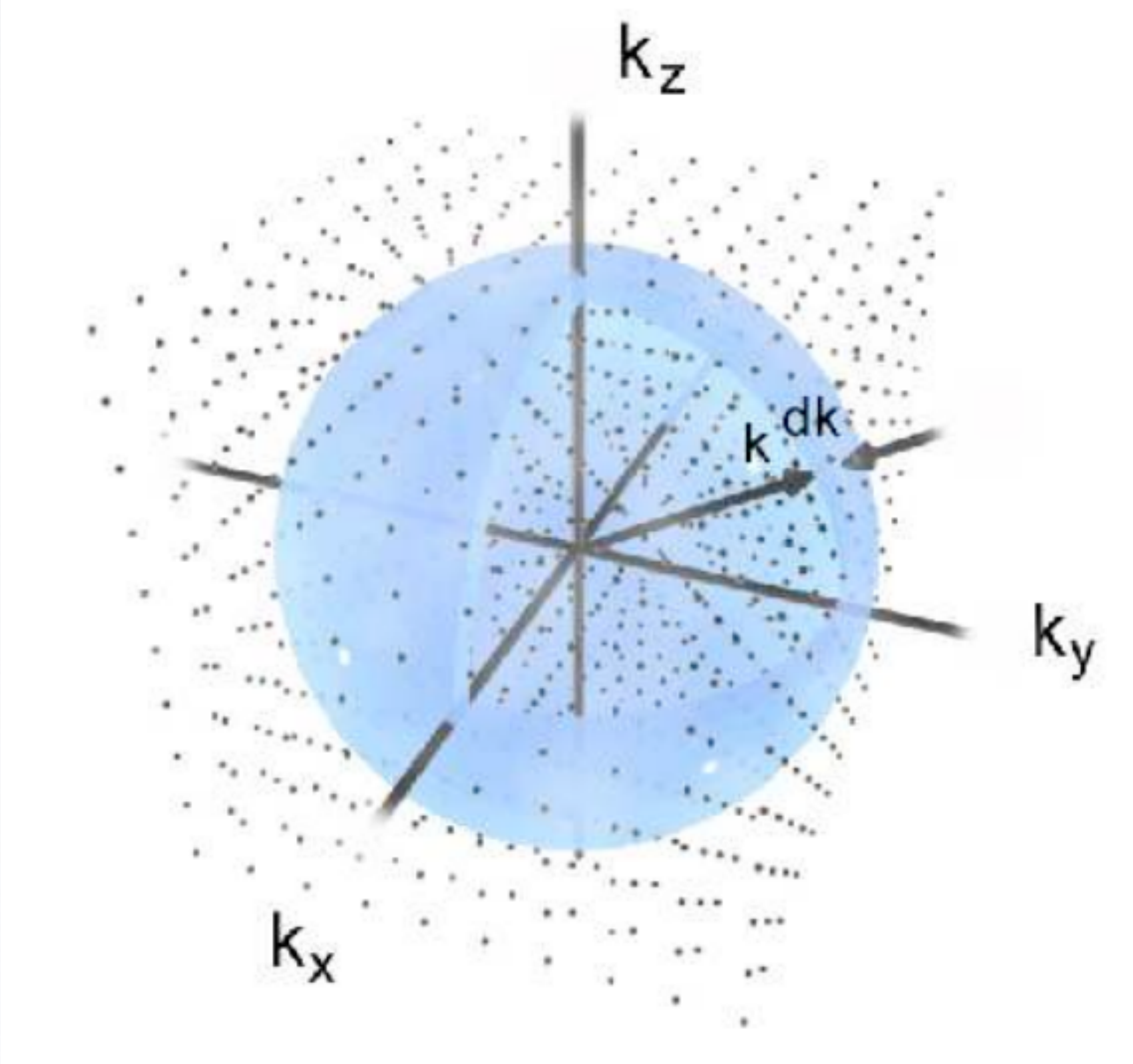


$$E_{\text{PW}} = \frac{\hbar^2(\vec{k}+\vec{G})^2}{2m} < E_{\text{cut}}$$



# 5. k-points





- Large real-space Wigner-Seitz cell => small Brillouin Zone (few k-points rqd)
- Small real-space Wigner-Seitz cell => large Brillouin Zone (many k-points rqd)

http://voh.chem.ucla.edu/vohtar/winter05/classes/115B/pdf/DFT_notes_Cocula_Carter.pdf

# 5. k-points

- The wavefunction (and energy) of each electron depends on both its quantum number *n* and its position *k* within the Brillouin Zone

$$\hat{h}_{\mathbf{k}}\, \psi_{n,\mathbf{k}}(\mathbf{r}) = \varepsilon_{n,\mathbf{k}}\, \psi_{n,\mathbf{k}}(\mathbf{r})$$

- Real-space quantities are computed by a **discrete sum over *n*** and **integration over *k* within the Brillouin Zone (approximated over a grid at finite *k*-points)**

- e.g. density, n(**r**)

$$n(\mathbf{r}) = \frac{V_{\text{cell}}}{(2\pi)^3} \int_{1\text{BZ}} \left( \sum_{n=1}^{N_{\text{el}}} |\psi_{n,\mathbf{k}}(\mathbf{r})|^2 \right) d\mathbf{k}$$

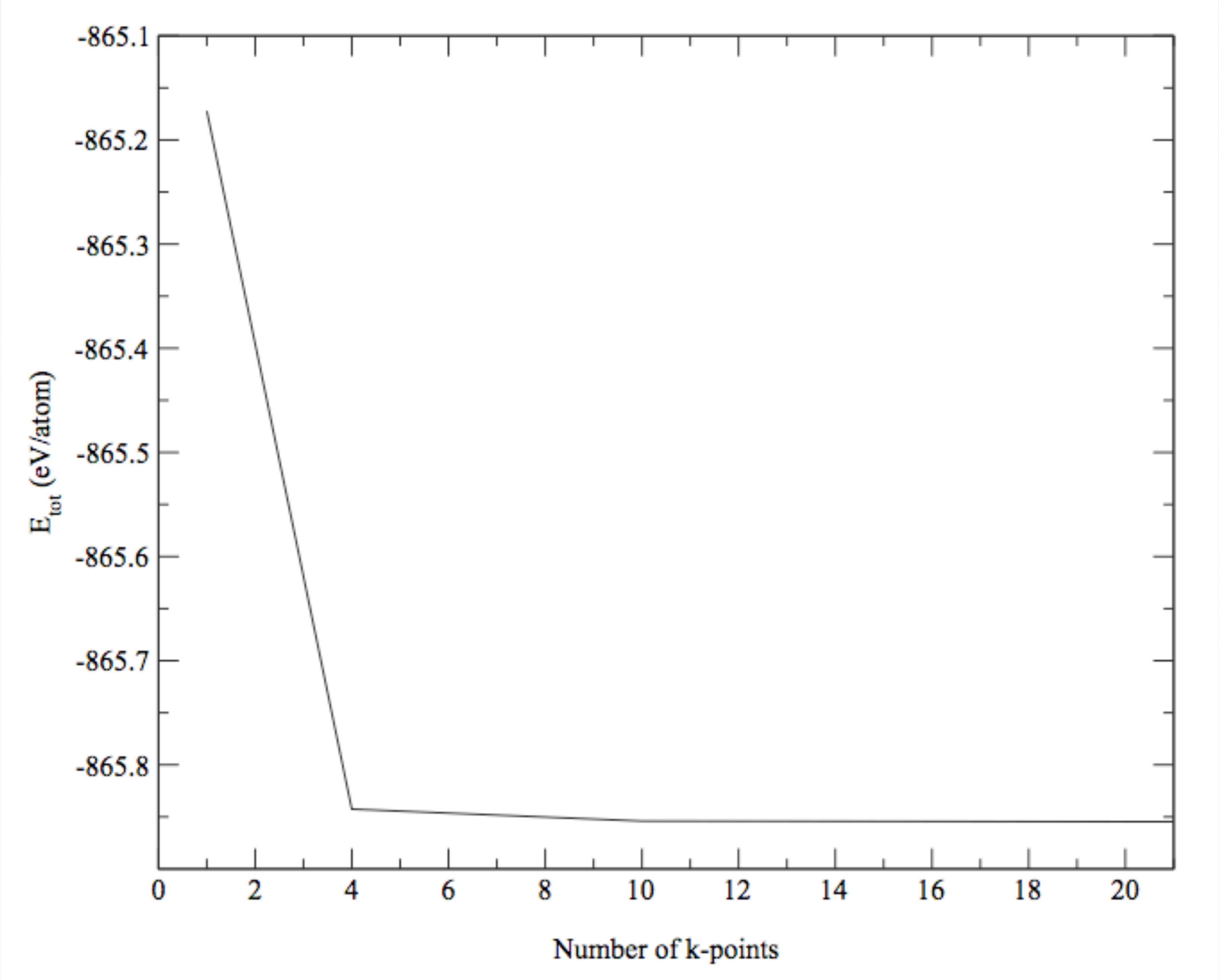
# 5. k-points + smearing

- The integration occurs over **occupied** electron states: energy ≤ Fermi energy
- At $T = 0$, equivalent to adding a step (Heaviside $H$) function:

$$\rho(\mathbf{r}) = \sum_n \frac{V_{\text{cell}}}{(2\pi)^3} \int_{1\text{BZ}} H(E_{\text{F}} - \varepsilon_{n,\mathbf{k}}) |\psi_{n,\mathbf{k}}|^2 d^3\mathbf{k}$$

- For a material with a band gap (no Fermi surface), no problem.
- For a *metal*, **huge** problem: this Heaviside function is **not smooth**
- **Solution:** replace Heaviside function with a smooth function. **Smearing.**
  - Fermi-Dirac occupancy at a *finite*, *artificial temperature*
  - Integral of a Gaussian centered at Fermi energy, with a finite width
  - Polynomial expansion (Methfessel-Paxton)
- **N.B.:** there is *another solution* called tetrahedron method for integrating *without smoothing*; however, it can have some issues that make it less than desirable for our purposes.

# k-points



http://voh.chem.ucla.edu/vohtar/winter05/classes/115B/pdf/DFT_notes_Cocula_Carter.pdf

# Summary

- What approximations do you control in your QE run?

1. Exchange-correlation (XC) functional
2. Pseudopotential
3. Basis set choice
4. Basis set energy cutoff
5. k-point sampling

# IV. Elastic constant calculations

# Calculating stress

- In a DFT calculation, you control where the atoms are:
  - periodic cell (including a "supercell")
  - atomic positions inside the cell
- This means that you apply *strain* to the cell, and can then compute the material response as *stress*.
- **This may seem backwards to you.**
- A DFT calculation can give you a total energy for the cell (potential energy) and derivatives of that energy.

$$\sigma_{ab} = \frac{1}{V}\frac{dE}{d\varepsilon_{ab}}$$

- Zero stress = equilibrium (or "relaxed") lattice constants
- Strain (infinitesimal change in the lattice vectors) affects
  - Lattice vectors and *reciprocal lattice vectors*
  - Planewave basis and kinetic energy changes

# Elastic constants $C_{ij}$

- Elastic constants (stiffnesses) are the *second derivative*

$$C_{abcd} = \frac{1}{V}\frac{d^2E}{d\varepsilon_{ab}d\varepsilon_{cd}} \qquad C_{ij} = \frac{1}{V}\frac{d^2E}{de_i de_j}$$

- Can be evaluated by fitting quadratic to energy vs. strain
- **Or** from derivative of stress with respect to strain

$$\frac{d\sigma_j}{de_i} = \frac{d}{de_i}\left(\frac{1}{V}\frac{dE}{de_j}\right) = \frac{1}{V}\frac{d^2E}{de_i de_j} - \frac{1}{V^2}\frac{dV}{de_i}\frac{dE}{de_j}$$

$$= C_{ij} - \frac{d\ln V}{de_i}\sigma_j \qquad \begin{cases} 1 & : i = 1,2,3 \\ 0 & : i = 4,5,6 \end{cases}$$

- Derivative evaluated using **finite difference**

$$\frac{f(x+h)-f(x)}{h} = f'(x) + \frac{1}{2}h\cdot f''(x) + \frac{1}{6}h^2\cdot f'''(x) + \cdots$$

$$\frac{f(x+h)-f(x-h)}{2h} = f'(x) + \frac{1}{6}h^2\cdot f'''(x) + \cdots$$

# Elastic constants $C_{ij}$: finite difference

- Introduce small ($\sim 10^{-3}$–$10^{-2}$) strains and evaluate stress

$$\frac{f(x+h)-f(x)}{h} = f'(x) + \frac{1}{2}h\cdot f''(x) + \frac{1}{6}h^2\cdot f'''(x) + \cdots$$

$$\frac{f(x+h)-f(x-h)}{2h} = f'(x) + \frac{1}{6}h^2\cdot f'''(x) + \cdots$$

- *Crystal symmetry* can make *f*'' = 0
- Small strains are susceptible to numerical errors in stress
- Large strains are susceptible to nonlinearities in stress
- Introduction of **finite strain** can
  - Lower symmetry (increase number of symmetry inequivalent k-points; slower calculation)
  - Lead to change in internal coordinates (relaxation)
  - Change the basis, including the number of planewaves

# Strains and volume changes

- Strain in real space "inversely" strains reciprocal space
  - Planewaves near cutoff move in and out with strain
  - Changes in volume change the *number* of planewaves
- Recommended approach:
  - **One set** of volume-changing strains ($e_1=e_2=e_3=x$)
  - **All others** conserve volume ($e_1=x$, $e_2=-x$, $e_3=?$)
  - Linear algebra to relate $C_{ij}$ to stress/strain derivatives