NSF

U.S. DEPARTMENT OF ENERGY

TACC
TEXAS ADVANCED COMPUTING CENTER

ODEN

Hands-on Mon.2

Phonon calculations with QUANTUM ESPRESSO

Paolo Giannozzi

Dept. Mathematics, Computer Science, Physics, and IOM-CNR
University of Udine, Italy

About QUANTUM ESPRESSO

- An open-source *distribution* (an integrated suite) of codes for electronic-structure calculations; organized into several packages
- Web site `http://www.quantum-espresso.org`, contains links to all relevant resources
- Documentation online at `https://www.quantum-espresso.org/documentation/` and in `Doc/`, `PW/Doc/`, `PHonon/Doc/` subdirectories
- Development takes place on `http://gitlab.com/QEF/q-e`
- Development funded by the QUANTUM ESPRESSO Foundation and by the European Union H2020 Project MaX - Materials at the Exascale

Practical phonon calculation in QUANTUM ESPRESSO

First step: scf calculation *at equilibrium positions* (performed by `pw.x`). Compute $n(\mathbf{r})$ and $\psi_{\mathbf{k},v}$ over the Irreducible Brillouin Zone for the symmetry group of the material.

- Single phonon calculation at finite wave-vector **q**
  - Generate $\psi_{\mathbf{k},v}$ and $\psi_{\mathbf{k}+\mathbf{q},v}$ in the Irreducible Brillouin Zone relative to the small group of **q**; Calculate $C(\mathbf{q})$, diagonalize, produce $\omega(\mathbf{q})$ and $U(\mathbf{q})$ (code `ph.x`)
- Single phonon calculation at Γ wave-vector (**q**=0)
  - Calculate $C(\mathbf{q}=0)$, diagonalize, produce $\omega(\mathbf{q}=0)$ and $U(\mathbf{q}=0)$ (code `ph.x`)
    For polar materials: calculate non-analytical terms that are missing from $C(\mathbf{q}=0)$ (LO-TO splitting are absent from $\omega(\mathbf{q}=0)$): specify option `epsil=.true.` to `ph.x` (will calculate and store in output file Z^* and ϵ^∞).
  - Impose Acoustic Sum Rule (ASR), add the nonanalytic LO-TO splitting, calculate cross sections (code `dynmat.x`)

Practical phonon calculation (2)

- Complete phonon dispersion calculation
  - Perform many single-phonon calculations on a uniform grid of wave-vectors $\mathbf{q}_i$, including $\mathbf{q} = 0$ (if system is polar, calculate in the latter case Z^* and ϵ^∞); save all $C(\mathbf{q}_i)$ (and Z^*, ϵ^∞) (code `ph.x` with option `ldisp=.true.`)
  - Perform inverse FFT of the $C(\mathbf{q}_i)$, obtain interatomic force constants in real space $C(\mathbf{R})$. For polar materials: a term having the same behavior for $\mathbf{q} \to 0$ as the non-analytic term is subtracted from $C(\mathbf{q}_i)$ before the Fourier Transform and re-added to $C(\mathbf{R})$, so that no problem related to the non-analytic behavior and related long-range character arises in the Fourier Transform (code `q2r.x`)
- Calculate phonons at any wave-vector, diagonalizing the dynamical matrix:

$$D_{st}^{\alpha\beta}(\mathbf{q}) = \frac{1}{\sqrt{M_s M_t}} \sum_{\mathbf{R}} C_{st}^{\alpha\beta}(\mathbf{R}) \exp(i\mathbf{q}\cdot\mathbf{R})$$

using code `matdyn.x`

Examples of this tutorial

- **Silicon** (LDA) in the diamond structure: a nonpolar insulator
  - Understand the procedure
- **AlAs** (LDA) in the zincblende structure: a polar insulator
  - Understand the importance of macroscopic electric fields
- **Pb** in the fcc structure: a metal, with important spin-orbit interactions
  - Understand the importance of a proper Fermi-surface sampling
  - Observe the effects of spin-orbit interactions

About parallelization

You may take advantage of parallelization of the `pw.x` and `ph.x` codes by using

```
ibrun -np N code-to-be-run
```

where `N` is the number of processors. Performs parallelization over plane waves (and real-space grids as well).

For small runs, it is not worth to set `N` larger than $\simeq 4-8$. You can use instead

```
ibrun -np N code-to-be-run -nk M
```

(alternatively: `-npool M`) that divides the `N` processors into `M` "pools", of `N/M` processors (must be an integer!). Parallelizing over plane waves takes place inside a pool, while parallelization over **k**-points takes place among pools.