Speeding-up the ground state Hamiltonian in real space

Jean Cauvin-Vila, Marc Torrent
CEA, DAM, DIF, Arpajon, France

Antoine Levitt
INRIA, CERMICS, Paris
PLANE-WAVE DFT HAMILTONIAN

\[ H\psi = \epsilon\psi \quad H = -\Delta + V_l + V_{nl} \]

\[ \psi \] is discretized on a plane-wave basis with a cut-off \( e_{cut} \)

In a plane-wave formalism:

- \(-\Delta\) is \( \frac{1}{G^2} \) \(\Rightarrow O(N_{pw})\)
- \(V_l\) is a convolution, applied as a FFT \(\Rightarrow O(N_{pw}ln(N_{pw}))\)
- \(V_{nl} = \sum |pi > Dij < pj| \) \(\Rightarrow O(N_{proj}N_{pw})\)

- Typical values : \(N_{pw} \sim 10^5\); \(N_{proj} \sim 2N_{atom} \ldots 32N_{atom}\)
- \(V_{nl}\) is a bottleneck on large systems!
The projectors are localized on the atoms!

Goal: compute $\langle p_j | \psi \rangle$ on the real space grid

Not so simple!

$\psi$ is only known on a coarse grid.
This is not so simple as to compute a 3D integral

$$\int_{|x-c|<r_c} \psi(x)p(x)dx$$

$p(r)$ is not limited in frequency.

Aliasing problem: $\frac{1}{N} \sum_r p(r) \Psi(r) \neq \sum_G p(G)\Psi(G)$
\[ I = \frac{1}{N} \sum_r p(r) \Psi(r) \]

\[ = \frac{1}{N} \sum_{j=0}^{N-1} p\left(\frac{j}{N}\right) \Psi\left(\frac{j}{N}\right) = \frac{1}{N} \sum_{j=0}^{N-1} \sum_{m \in \mathbb{Z}} e^{in2\pi j/N} p(2\pi m) \sum_{n \in [-n_{\text{max}}, n_{\text{max}}]} \Psi(2\pi n) e^{in2\pi j/N} \]

\[ = \frac{1}{N} \sum_{m \in \mathbb{Z}} \sum_{n \in [-n_{\text{max}}, n_{\text{max}}]} p(2\pi m) \Psi(2\pi n) \sum_{j=0}^{N-1} (e^{i(m+n)2\pi/N})^j \]

\[ = \sum_{k \in \mathbb{Z}, \Omega^*} p(g-kN)\Psi(g) = \sum_{g \in \Omega^*} p(g)\Psi(g) + \sum_{k \neq 0 \in \mathbb{Z}, \Omega^*} p(g-kN)\Psi(g) \]

- First idea: cut \( p \) in frequency (project on the PW basis)
  Not a good idea because we loose localization in real space
- We search for a accuracy/locality compromise
1. Densify the real space grid
   i.e.: add plane wave components for $p$ and $\psi$: $(e_{\text{cut}} \rightarrow e_{\text{max}})$

2. Put zero in the these new components for $\psi$

3. Optimize the new $p$ components in order to localize the projector a much as possible
   $\Rightarrow$ $p$ support is enlarged but not too much!

\[ I = \frac{1}{N} \sum_{r < \alpha r_c} p(r) \Psi(r) \]
\[ = \sum_{g^2 < e_{\text{cut}}^2} p(g) \Psi(g) + \sum_{e_{\text{cut}}^2 < g^2 < e_{\text{max}}^2} p(g) \Psi(g) \]

\[ \text{optimized} \quad \text{zero} \]
1. Fix $p(g)$ for $\frac{1}{2} g^2 \leq e_{cut}$

2. Set $p(g) = 0$ for $\frac{1}{2} g^2 > \frac{1}{2} G_{\text{max}} \approx e_{cut}$
   
   $p(r)$ support is enlarged: $r \leq r_c \rightarrow r \leq \alpha r_c$

3. Optimize $p(g)$ for $e_{cut} \leq \frac{1}{2} g^2 \leq \frac{1}{2} G_{\text{max}}$

   Minimize $I = \int_{\alpha r_c}^{\infty} r^2 p(r) dr$

   i.e. solve a least squares problem
Several improvements proposed:


Our proposal:
Add a degree of freedom for the optimization of $p$

$$Err = | < p, \Psi >_{\Omega^*} - < p_{opt}, \Psi >_{\Omega^*} |$$

$$Err \leq | < p, \Psi >_{\Omega^*} - < p_{opt}, \Psi >_{\Omega^*} | + | < p_{opt}, \Psi >_{\Omega} - < p_{opt}, \Psi >_{\Omega^*} |$$

$$Err \leq \lambda \sqrt{\int_{0}^{G_{\text{max}}} |p - p_{opt}|^2 w(q) dq} + \beta \sqrt{\int_{\alpha \Gamma_c} |p_{opt}|^2 dr}$$

Not considered in KS method

Frequency distribution of $\psi$
FIRST IMPLEMENTATION IN ABINIT

- Modification of `pawtab` and `hamiltonian` datastructures
  `rs_proj` class added
- `nonlop` flowchart changed

\[ \langle p_j | \psi \rangle \text{ in real space} \]
\[ \langle r | V_{nl} | \psi \rangle = \sum \langle r | p_i \rangle D_{ij} \langle p_j | \psi \rangle \]

**Components of total free energy (in Hartree):**

- **Kinetic energy** = 8.01762316556784E+00
- **Hartree energy** = 1.439333912141E+00
- **XC energy** = -3.69768237303331E+00
- **Ewald energy** = -1.2786412110521E+01
- **PspCore energy** = 5.41017918797015E+01
- **Loc. psp. energy** = -6.4527427317843E+00
- **Spherical terms** = 2.5291503734109E+00
- **Total** = -1.04321528000365E+01

**Components of total free energy (in Hartree):**

- **Kinetic energy** = 8.01762316556784E+00
- **Hartree energy** = 1.439333912141E+00
- **XC energy** = -3.69768237303331E+00
- **Ewald energy** = -1.2786412110521E+01
- **PspCore energy** = 5.41017918797015E+01
- **Loc. psp. energy** = -6.4527427317843E+00
- **Spherical terms** = 2.5291503734109E+00
- **Total** = -1.04321528000365E+01

"Double-counting" decomposition of free energy:

- **Band energy** = 2.2448285542259E+00
- **Ewald energy** = -1.2786412110521E+01
- **PspCore energy** = 5.41017918797015E+01
- **Dble-C XC-energy** = -2.99873574296493E+01
- **Spherical terms** = -1.36519811113875E+01
- **Total (DC)** = -1.04321528000365E+01

**Toy model:** Diamond
\[ ecut = 15 \text{ Ha} \]
- Add precision control
- Add parallelism (grid points, projector indexes)
- Implement forces and stress tensor
- Mix with the application of the local operator

Idea: add a pointer to function in fourwf

\[ func \rightarrow V_{local} \text{ (standard)} \]
\[ func \rightarrow V_{local} + V_{nl} \]

Target application: molecular dynamics