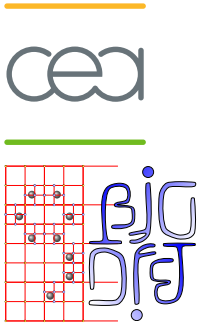


Numerical Methods in Density Functional Theory

TU UNIVERSITY - BERLIN

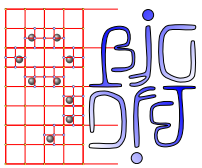
Poisson Solvers with Interpolating Scaling Functions for Electronic Structure Calculations

Luigi Genovese

L_Sim - CEA Grenoble

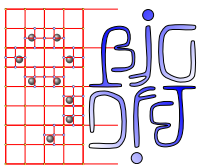
23 July 2008

Outline



1. Self-Consistent potential in DFT
   - Poisson Solver
2. Poisson Solver for Free BC
   - Calculation of the Poisson Kernel
   - Performances
3. A Poisson solver for Surfaces BC
   - Performances
4. Summary and outlook

Self-Consistent potential in DFT



In the DFT calculation in the Kohn-Sham formalism we have:

Find a set of orthonormal orbitals $\Psi_i(\mathbf{r})$ that minimizes:

$$E = -\frac{1}{2}\sum_{i=1}^{N/2}\int \Psi_i^*(\mathbf{r})\nabla^2\Psi_i(\mathbf{r})d\mathbf{r} + \frac{1}{2}\int \rho(\mathbf{r})V_H(\mathbf{r})d\mathbf{r} + E_{xc}[\rho(\mathbf{r})] + \int V_{ext}(\mathbf{r})\rho(\mathbf{r})d\mathbf{r}$$

where

$$\rho(\mathbf{r}) = 2\sum_{i=1}^{N/2}\Psi_i^*(\mathbf{r})\Psi_i(\mathbf{r})$$

$$\nabla^2 V_H(\mathbf{r}) = -4\pi\rho(\mathbf{r})$$

Operations performed in BigDFT code

Different operators

Daubechies:

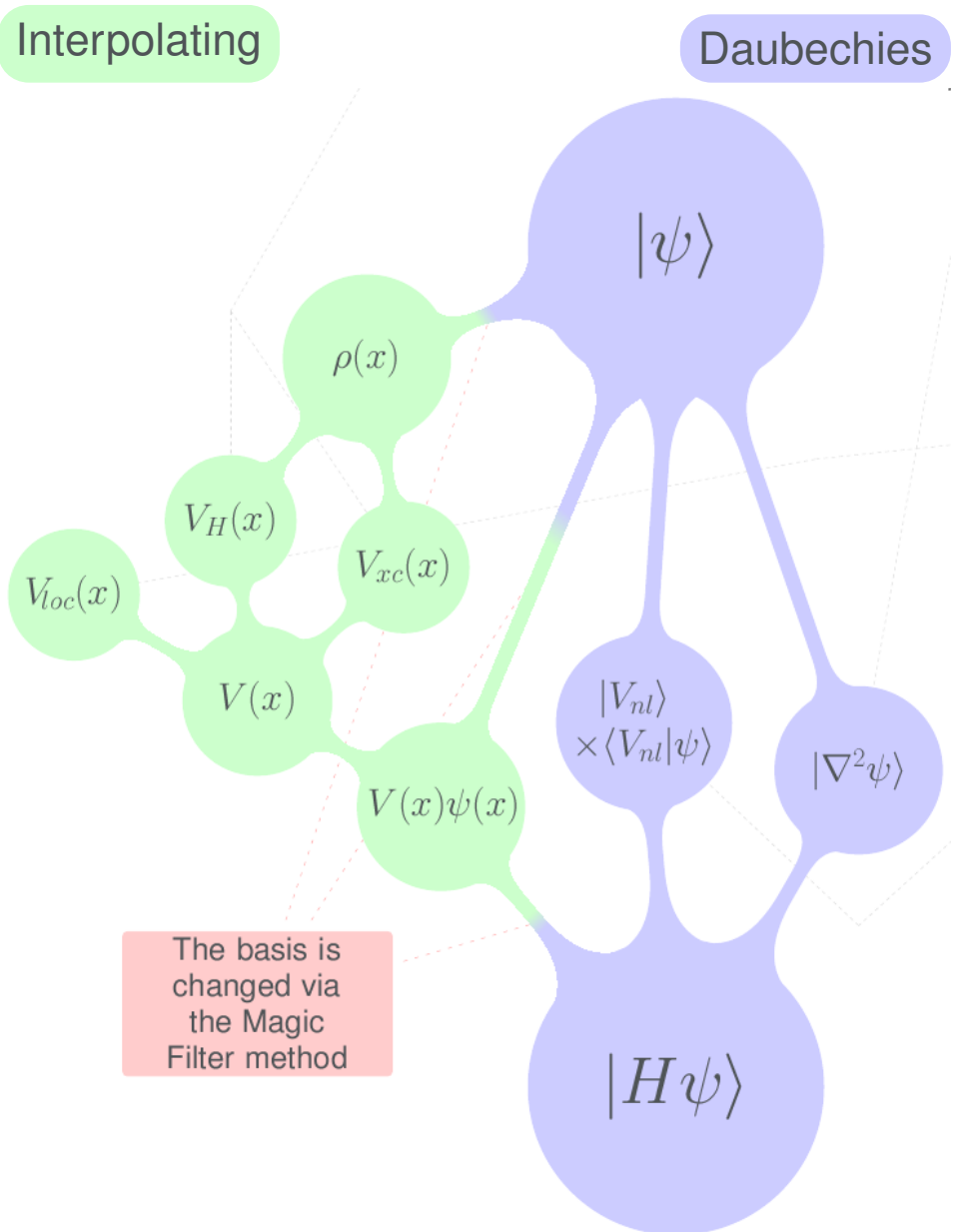
- Kinetic energy
- Scalar products, V_{nl}

Interpolating SF:

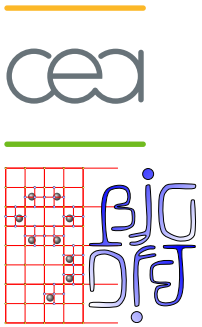
- V_{loc} (gaussians)
- XC (ρ needed)
- Hartree : $\nabla^2 V_H = \rho$

Numerical operations

- Convolutions with short filters
- Scalar products
- FFT (Poisson Solver)



The Poisson Solver in electronic structure calculation



During the minimization procedure we need to perform

Poisson's equation

Calculation of the self-consistent potential:

$$\nabla^2 V_H(\mathbf{r}) = -4\pi\rho(\mathbf{r})$$

Such equation should be solved at each minimisation iteration. Need of having an efficient and accurate formalism.

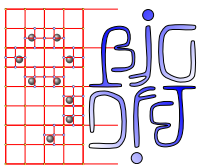
Plane waves approach

The most common approach. Uses the fourier components

$$f(x,y,z) = \sum_{p_x,p_y,p_z} e^{-2\pi i\left(\frac{p_x}{L_x}x+\frac{p_y}{L_y}y+\frac{p_z}{L_z}z\right)} f_{p_x,p_y,p_z}$$

The Poisson equation is algebraic in the Fourier coefficients

Poisson Solver with plane waves treatment



The Laplacian is diagonal in Plane waves representation

Immediate solution

$$V_{p_x,p_y,p_z} = \frac{1}{\pi} \frac{1}{\left(\frac{p_x}{L_x}\right)^2 + \left(\frac{p_y}{L_y}\right)^2 + \left(\frac{p_z}{L_z}\right)^2} \rho_{p_x,p_y,p_z} \ ,$$

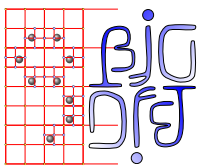
Characteristics

- Simple and fast, easy to parallelize (FFT)
- Automatically implement Periodic BC on a finite volume
- Do not fix the value of $V_{0,0,0}$

$\Rightarrow$ May result in problems for systems with other BC

How to solve this equation for other BC?

Problems with Plane Wave expansion



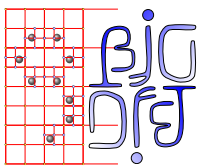
How to remove long-distance interactions?

- Model their effect and subtract it
  - Good for integrated quantities (e.g. total energy) but still inefficient for the local values $V(\mathbf{r})$
  - The size L of the effective system must be enlarged wrt the original one
- Modify the kernel operator $K = K_{\text{short}} + K_{\text{long}}$
  - Does not implement well short-distance behaviour, resulting in errors
  - These errors decrease when the size of the system is large

In both cases

We must consider a size that is larger than the size of the original system

Isolated BC: the Green function treatment



Consider the Poisson equation for isolated BC. In this case the solution is given by

Green's function for Free BC (kernel)

$$\nabla^2 \frac{1}{r} = -4\pi\delta(r) \Longrightarrow V(\mathbf{r}) = \int \mathrm{d}\mathbf{r}' \frac{\rho(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}$$

An unambiguous solution

Such prescription is unique and is compatible with Free BC.

How to implement it?

Plane-wave based approaches:

- Truncated kernel, numerically or analytically
- Screening Functions

Approximate treatment, large box required.

Need of an accurate and efficient algorithm

Poisson Solver with Interpolating Scaling Functions

A convenient basis for an electrostatic problem

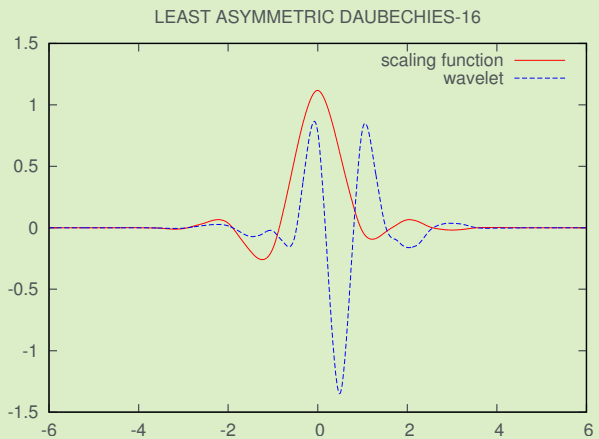
Interpolating Scaling Functions

A set of localised functions centered on the nodes of a uniform mesh

$$\varphi_j(x) = \varphi_0(x-j)$$

Undergo multiscale relation

$$\varphi(x) = \sum_{j=-m}^{m} \underbrace{h_j}_{\text{filters}} \varphi(2x-j)$$



- ✔ The expansion coefficients are the real space values $\rho_{j_x,j_y,j_z} = \rho(h_x j_x, h_y j_y, h_z j_z)$
- ✔ Represents exactly an order m polynomial
- ✔ The first m discrete and continuous moments coincide

A finite three-dimensional convolution

The expression of the potential in this basis is thus intuitive:

$$V(\mathbf{i}) = \sum_{\mathbf{j}} K_{\mathbf{ij}}\, \rho_{\mathbf{j}}$$

Where the central object is the

Poisson Kernel in the ISF basis

$$K_{\mathbf{ij}} = K_{\mathbf{i}-\mathbf{j}}\,, \qquad K_{\mathbf{i}} = \int K(|\mathbf{r}|)\varphi_{\mathbf{i}}(\mathbf{r})d\mathbf{r}\,, \quad K(r) = \frac{1}{r}$$

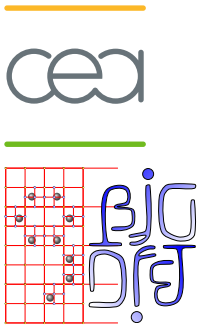
Values of the potential are obtained via a convolution

$$V(\mathbf{i}) = \sum_{\mathbf{j}} K_{\mathbf{i}-\mathbf{j}}\, \rho_{\mathbf{j}}$$

It can be treated via a zero-padded FFT algorithm

- ✔ Exact, easy to parallelize (FFT)
- ✔ For a box of N^3 points, it reduces the scaling from $O(N^6)$ to $O(N^3 \log N)$

Characteristics of the approach



This approach is:

- Explicit, guarantees the good BC
- Real space-based, immediate interpretation of the expansion coefficients
- Can be combined with other real-space treatments of the density (e.g. XC) combined with ABINIT XC routines
- Can be used independently from the BigDFT code
- Preserves the first m (multipole) moments of the electrostatic potential

Requires only the evaluation of the kernel

We need to evaluate N^3 integrals $K_{\mathbf{i}} = \int K(|\mathbf{r}|)\varphi_{\mathbf{i}}(\mathbf{r})d\mathbf{r}$ where $\varphi_{\mathbf{i}}(\mathbf{r}) = \varphi_{i_x}(x)\varphi_{i_y}(y)\varphi_{i_z}(z)$

ISF basis is a tensor product

Gaussian tensor product decomposition

It can be shown that (Beylkin et *al.*)

Approximation with gaussians

$$\frac{1}{r} \simeq \sum_k \omega_k e^{-p_k r^2}$$

with $k = 1, \cdots, 89$, p_k , ω_k suitably chosen

Accuracy of 10^{-8} for $r \in (10^{-9}, 1)$

We can rescale for $R \in (0, L)$.

The computational cost is reduced $N^3 \longrightarrow 89 \times N$

$$K_{\mathbf{j}} = \sum_{k=1}^{89} \omega_k K_{j_x}(p_k) K_{j_y}(p_k) K_{j_z}(p_k)$$

$$K_j(p) = \int \varphi_0(x) e^{-p(x-j)^2} dx$$

Other properties of the scaling functions

The computational cost is reduced $N^3 \longrightarrow 89 \times N$.

Moreover

The scaling property of the interpolets

$$\varphi_0(x/2) = \sum_j h_j \varphi_j(x) \ ,$$

Implies similar condition of the one-dimensional function

$$K_j(4p) = \frac{1}{2} \sum_j \mathrm{h}_j K_{2i-j}(p) \ .$$

Thus we can evaluate the integrals for low p (not too sharp gaussians), then rescaling.

ISF properties allows us to gain in accuracy

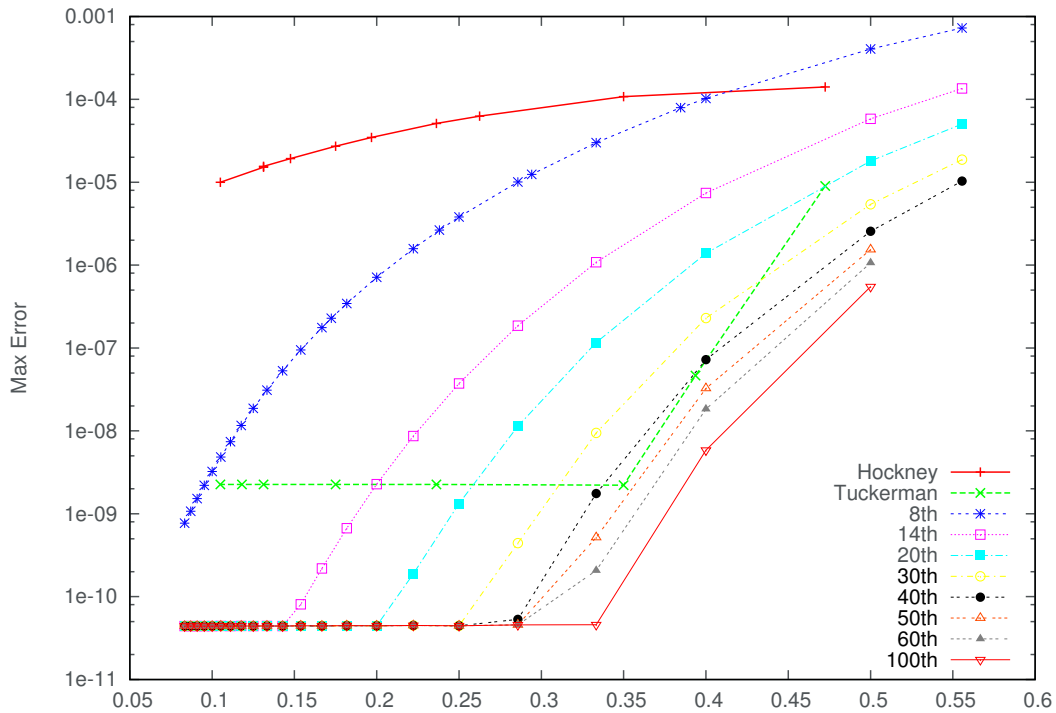
Characteristics

Very fast with moderate memory occupation:

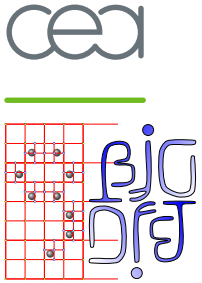
Elapsed Time for a 128^3 grid on a Cray XT3

proc	1	2	4	8	16	32	64
s	.92	.55	.27	.16	.11	.08	.09

More precise than other existing Free BC Poisson Solvers



Characteristics of the Solver



In summary, we have developed a technique

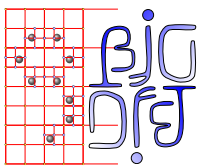
- Free boundary conditions
- Very high accuracy
- Good computational performance, easy to parallelize
- Can be used also in other contexts and/or combined with other treatment (e.g. XC)
- Coupled with ABINIT XC routines

In BigDFT code, for big systems it represents a small amount (typical values are less than 2-3% for big systems) of the overall computation

L. Genovese, T. Deutsch, A. Neelov, S. Goedecker, G. Beylkin

J. Chem. Phys. **125**, 074105 (2006)

The Surfaces boundary conditions



The same formalism can be applied to other BC

Surfaces BC

A domain isolated in one direction (say y) and periodic in x and z, with periods L_x and L_z. A function f which lives in such a domain can be expanded as

$$f(x,y,z) = \sum_{p_x,p_z} e^{-2\pi i\left(\frac{p_x}{L_x}x+\frac{p_z}{L_z}z\right)} f_{p_x,p_z}(y)$$

without any loss of generality.

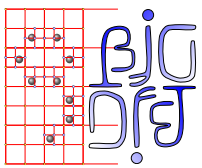
Mixed representation

For such functions the Poisson's equation become

$$\left(\partial_y^2 - \mu^2_{p_x,p_z}\right) V_{p_x,p_z}(y) = \rho_{p_x,p_z}(y)\ ,$$

where $\mu^2_{p_x,p_z} = 4\pi^2(p_x/L_x)^2 + (p_z/L_z)^2$.

A Green's function formalism



The Green's function for the one-dimensional Helmoltz equation can be used

$$\left(\partial_y^2 - \mu^2\right) G(\mu; y) = \delta(y) \, ;$$

$$G(\mu; y) = \begin{cases} -\frac{1}{2\mu} e^{-\mu|y|} & \mu > 0 \\ \frac{1}{2}|y| & \mu = 0 \end{cases} ,$$

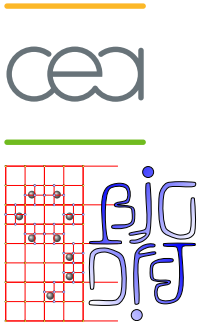
the components of the potential can be carried out:

A Green's function for each Fourier component

$$V_{p_x,p_z}(y) = \int \mathrm{d}y' \, G(\mu_{p_x p_z}; y - y') \rho_{p_x,p_z}(y') \, .$$

We can use Interpolating Scaling Functions for the Isolated direction.

Surfaces BC with Mixed representation



By defining the one-dimensional kernels

$$K(\mu_{p_x,p_z};j) = \int G(\mu_{p_x p_z};y)\varphi_j(y)\mathrm{d}y$$

We can have a treatment similar to the Free BC case

1-dim convolution for each reciprocal space component

$$V_{p_x,p_z}(i) = \sum_j K(\mu_{p_x,p_z};i-j)\rho_{p_x,p_z}(j)$$

From mixed representation to full real space

The calculation can be performed with a (semi-)zero-padded FFT algorithm. The I/O are function of the real domain.

Speed-up the calculation

The calculation of the kernel can be improved

Speed

The analytic form of the Green functions allows for recursion relations

$\longrightarrow$ improvement in speed of a factor of $(N_z + 1)/3$

Accuracy

The scaling relation will again help in the kernel

$$K(2\mu; j) = \frac{1}{2} \sum_j \mathrm{h}_j K(2\mu, 2i - j)$$

A "Free-Lunch" case

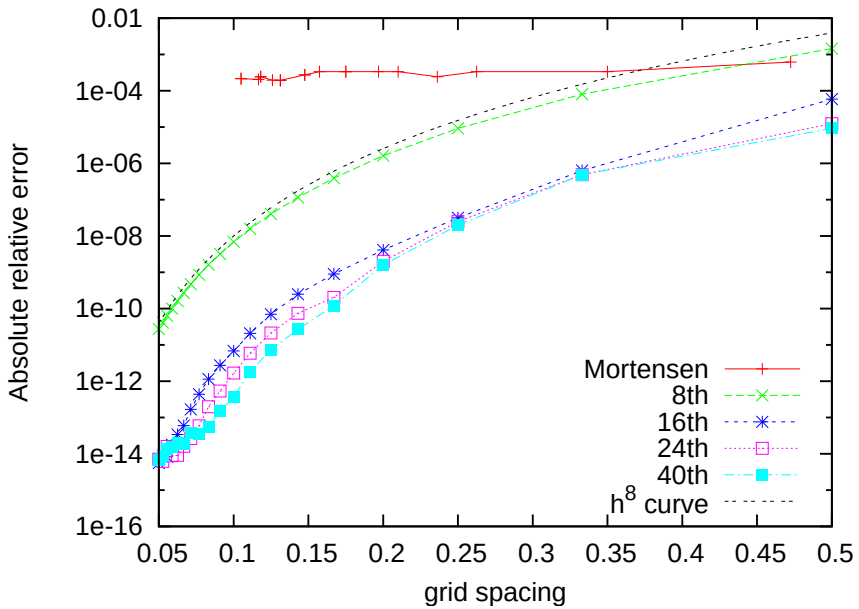
We have the same advantages of the Free BC treatment with an explicit formalism conceived for surfaces BC

Poisson Solver for surface boundary conditions

Elapsed Time on a Cray XT3, 128^3 grid

#proc	1	2	4	8	16	32	64
sec	.43	.26	.16	.10	.07	.05	.04

More precise than other treatments



L. Genovese, T. Deutsch, S. Goedecker

J. Chem. Phys. **127**, 054704 (2007)

A Poisson solver for surface problems

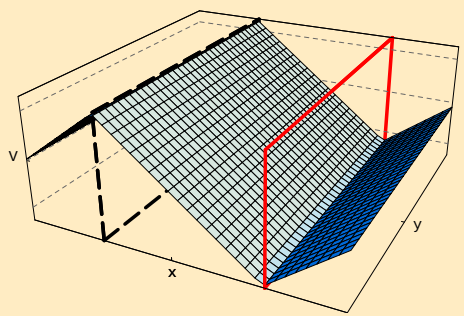
Like the Free BC case

We developed a technique

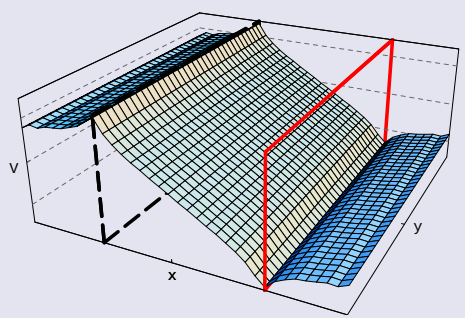
- ✔ Accurate and fast, easy to parallelize
- ✔ Can be applied both in real or reciprocal space codes
- ✔ Explicit treatment No supercell or screening functions
- ✔ More precise than other existing approaches
- ✔ Allows comparisons betweeen different backgrounds (charged systems)

Example of the plane capacitor:

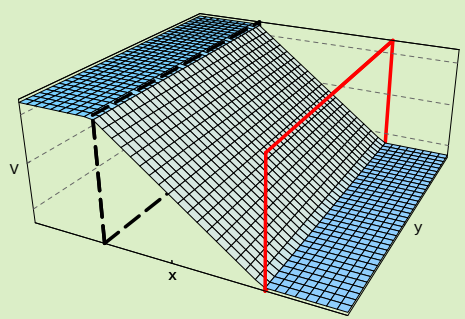
Periodic



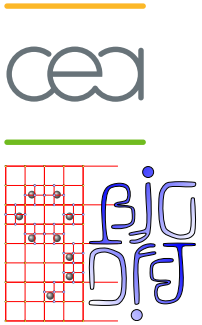
Hockney



Our approach

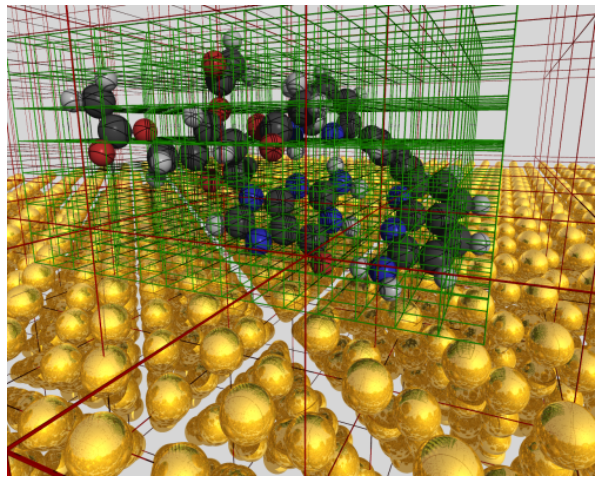


Flexibility: the Poisson Solver in BigDFT



Flexible Boundary Conditions

- Direct approach (non-iterative)
- Explicit, high precision



Clean and precise treatment of the electrostatic

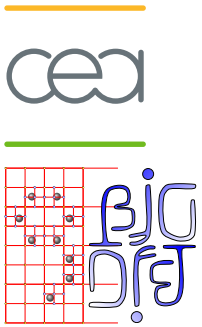
Several applications:

- Impurities, dopants
- Surface properties (adsorbtion, cathalysis)
- *Charged* defects, in bulk or in surfaces

These Solvers can be used independently from BigDFT code

Already integrated in other DFT codes (CP2K, ABINIT, OCTOPUS)

Summary and outlook



Interpolating SF can be use to solve the Poisson's equation

$$\nabla^2 V_H = -4\pi\rho$$

with a Green's function treatment, in different environements:

Isolated BC

$$V_H(\vec{j}) = \int \mathrm{d}\vec{x}\, \frac{\rho(\vec{x})}{|\vec{x}-\vec{j}|}$$

Surfaces BC

$$V_{p_x,p_z}(y) = \int \mathrm{d}y'\, G(\mu_{p_x p_z}; y-y')\rho_{p_x,p_z}(y')$$

From $\rho(\vec{j})$ on a uniform grid

Explicit treatment of the boundary conditions

Can be generalised to other environments

- Wires BC
- Spherical heterostructures (multipole expansion)