Analysis, post-processing and visualization tools

Javier Junquera

Andrei Postnikov
Summary of different tools for post-processing and visualization

DENCHAR

PLRHO

DOS, PDOS

MACROAVE

SIESTA’s output with XCRYSDEN
Summary of different tools for post-processing and visualization

- **DENCHAR**
- **PLRHO**
- **DOS, PDOS**
- **MACROAVE**
- **SIESTA’s output with XCRYSDEN**
DENCHAR plots the charge density and wave functions in real space

Wave functions

\[ \psi_i(\vec{k}, \vec{r}) = \sum_{\mu} e^{i\vec{k} \cdot \vec{R}_\mu} c_{\mu i}(\vec{k}) \phi_\mu(\vec{r}) \]

Coefficients of the eigenvector \( \psi_i(\vec{k}) \) with eigenvalue \( E_i(\vec{k}) \)

atomic orbitals

Charge density

\[ \rho(\vec{r}) = \sum_i \int_{BZ} n_i(\vec{k}) | \psi_i(\vec{k}, \vec{r}) |^2 d\vec{k} \]

\[ = \sum_{\mu\nu} \rho_{\mu\nu} \phi^*_\nu(\vec{r}) \phi_\mu(\vec{r}) \]

density matrix
DENCHAR operates in two different modes: 2D and 3D

2D

• Charge density and/or electronic wave functions are printed on a regular grid of points contained in a 2D plane specified by the user.
• Used to plot contour maps by means of 2D graphics packages.

3D

• Charge density and/or electronic wave functions are printed on a regular grid of points in 3D.
• Results printed in Gaussian Cube format.
• Can be visualized by means of standard programs (Moldel, Molekel, Xcrysden)
How to compile DENCHAR...

```
$ cd siesta/Src
$ make denchar
ifort -c -O3  f2kcli.F90
ifort -c -O3  m_denchar_init.F
ifort -c -O3  m_denchar_geom.f
ifort -c -O3  m_denchar_io.F
ifort -c -O3  m_denchar_neighb.f
ifort -c -O3  m_denchar_work.f
ifort -c -O3  denchar.F
ifort -o denchar \ 
    m_denchar_init.o m_denchar_geom.o m_denchar_io.o m_denchar_neighb.o
    m_denchar_work.o denchar.o precision.o recipes.o f2kcli.o bessph.o chkdim.o
dismin.o dot.o iodm.o memory.o paste.o radfft.o i.o.o spatial.o volcel.o
parallel.o parallelsubs.o memoryinfo.o sys.o listsc.o atmparms.o atmfuncs.o
atm_types.o m_memory.o radial.o spher_harm.o basis_io.o basis_types.o
pseudopotential.o chemical.o xml.o files.o bsd.o pxf.o \
    libfdf.a linalg.a
```

Use the same arch.make file as for the compilation of serial SIESTA

Versions before 2.0.1, please check for patches in
http://fisica.ehu.es/ag/siesta-extra/issues.html

...and where to find the User’s Guide and some Examples

```
$ cd siesta/Util/Denchar
$ ls
Docs Examples README-Source
```
How to run DENCHAR...

SIESTA

WriteDenchar .true.
WriteWaveFunctions .true.
%block WaveFuncKPoints
  0.0 0.0 0.0
%endblock WaveFuncKPoints

Output of SIESTA required by DENCHAR

SystemLabel.PLD
SystemLabel.DIM
SystemLabel.DM
SystemLabel.WFS (only if wave functions)
ChemicalSpecies.ion (one for each chemical species)

DENCHAR

$ ln –s ~/siesta/Src/denchar .
$ denchar < dencharinput.fdf

You do not need to rerun SIESTA to run DENCHAR as many times as you want
Input of DENCHAR
General issues

• Written in fdf (Flexible Data Format), as in SIESTA

• It shares some input variables with SIESTA
  
  SystemLabel
  NumberOfSpecies
  ChemicalSpeciesLabel

• Some other input variables are specific of DENCHAR (all of them start with “Denchar.”)
  
  To specify the mode of usage
  To define the plane or 3D grid where the charge/wave functions are plotted
  To specify the units of the input/output

• Input of DENCHAR can be attached at the end of the input file of SIESTA
Input of DENCHAR
How to specify the mode of run

• **Denchar.TypeOfRun**  (string)  2D or 3D

• **Denchar.PlotCharge**  (logical)  .TRUE. or .FALSE.
  If .true. SystemLabel.DM must be present

• **Denchar.PlotWaveFunctions**  (logical)  .TRUE. or .FALSE.
  If .true. SystemLabel.WFS must be present

Either one or the other (or both of them) must be .true.
Input of DENCHAR
How to specify the plane

- Denchar.PlaneGeneration (string)
  - NormalVector
  - TwoLines
  - ThreePoints
  - ThreeAtomicIndices

+ more variables to define the
  generation object (the normal vector, lines, points or atoms)
  origin of the plane
  x-axis
  size of the plane
  number of points in the grid

- Different variables described in the User Guide
  (take a look to the Examples)
Output of DENCHAR 2D mode

Charge density

Spin unpolarized:
  self-consistent charge (.CON.SCF)
  deformation charge (.CON.DEL)

Spin polarized:
  density spin up (.CON.UP)
  density spin down (.CON.DOWN)
  deformation charge (.CON.DEL)
  magnetization (.CON.MAG)

Wave functions

Wave function for different bands
(each wavefunction in a different file)
  .CON.WF#, where # is the number of the wf
(If spin polarized, suffix .UP or .DOWN)

Format

<table>
<thead>
<tr>
<th>$x_i$</th>
<th>$y_i$</th>
<th>$f(x_i, y_i)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>-5.00000</td>
<td>-5.00000</td>
<td>-0.00560</td>
</tr>
<tr>
<td>-5.00000</td>
<td>-4.74359</td>
<td>-0.00402</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Output of DENCHAR 3D mode

Charge density
Spin unpolarized:
self-consistent charge (.RHO.cube)
deformation charge (.DRHO.cube)
Spin polarized:
density spin up (.RHO.UP.cube)
density spin down (.RHO.DOWN.cube)
deformation charge (.DRHO.cube)

Wave functions
Wave function for different bands
(each wavefunction in a different file)
.WF#.cube, where # is the number of the wf
(If spin polarized, suffix .UP or .DOWN)

Format

Gaussian Cube format
Atomic coordinates and grid points in the reference frame given in the input
Reference frame orthogonal
Summary of different tools for post-processing and visualization

**DENCHAR**

**PLRHO**

**DOS, PDOS**

**MACROAVE**

**SIESTA’s output with XCRYSDEN**
PLRHO plots a 3D isosurface of the charge density and colours it with a second function

- $\rho (\vec{r})$
- LDOS integrated in a given energy interval
- $\rho (\vec{r}) +$ electrostatic potential
- $\rho (\vec{r}) +$ total potential
- $\rho (\vec{r}) +$ spin density

Plrho reads the values of the functions in the real space grid and interpolates to plot the 3D surface.
How to compile PLRHO

• First you need to install the PGPLOT library, available from
  http://www.astro.caltech.edu/~tjp/pgplot

• You can find plrho at
  ~/siesta/Utils/Plrho

• Then compile PLRHO with
  $ f90 plrho.f –lX11 –lpgplot –o plrho

• Check plrho_guide.txt for extra information.
How to run PLRHO

SIESTA

SaveRho .true.
SaveElectrostaticPotential .true.
SaveTotalPotential .true.
%block LocalDensityOfStates
%block AtomicCoordinatesOrigin

Output of SIESTA required by PLRHO

SystemLabel.RHO
SystemLabel.VH
SystemLabel.VT
SystemLabel.LDOS

PLRHO

Prepare the input file plrho.dat
$ plrho

Depending on what you want to plot

If you want to center the system

You do not need to rerun SIESTA to run PLRHO as many times as you want
### Input of PLRHO: plrho.dat

<table>
<thead>
<tr>
<th>'h2o'</th>
<th>System label, used to name input files</th>
</tr>
</thead>
<tbody>
<tr>
<td>'vh'</td>
<td>Function(s) to plot ('rho'</td>
</tr>
<tr>
<td>90.0 0.0 -90.0</td>
<td>Euler rotation angles alpha, beta, gamma</td>
</tr>
<tr>
<td>4.0e-3</td>
<td>Value of electron density (or LDOS) for isosurface</td>
</tr>
<tr>
<td>-0.08 +0.01 +0.12</td>
<td>Saturation range for color function</td>
</tr>
<tr>
<td>'unformatted'</td>
<td>Data format ('formatted'</td>
</tr>
<tr>
<td>'/CPS'</td>
<td>Output option ('/XWIN'</td>
</tr>
</tbody>
</table>
Input of PLRHO: plrho.dat

<table>
<thead>
<tr>
<th>'h2o'</th>
<th>System label, used to name input files</th>
</tr>
</thead>
<tbody>
<tr>
<td>'v:h'</td>
<td>Function(s) to plot ('rho'</td>
</tr>
<tr>
<td>90.0 0.0 -90.0</td>
<td>Euler rotation angles alpha, beta, gamma</td>
</tr>
<tr>
<td>4.e-3</td>
<td>Value of electron density (or LDOS) for isosurface</td>
</tr>
<tr>
<td>-0.08 +0.01 +0.12</td>
<td>Saturation range for color function</td>
</tr>
<tr>
<td>'unformatted'</td>
<td>Data format ('formatted'</td>
</tr>
<tr>
<td>'/CPS'</td>
<td>Output option ('/XWIN'</td>
</tr>
</tbody>
</table>

'rho' $\rho (\vec{r})$

'ldos' LDOS integrated in a given energy interval

'vh' $\rho (\vec{r}) +$ electrostatic potential

'vt' $\rho (\vec{r}) +$ total potential

'spin' $\rho (\vec{r}) +$ spin density
Input of PLRHO: plrho.dat

- 'h2o': System label, used to name input files
- 'vh': Function(s) to plot ('rho'|'ldos'|'spin'|'vt'|'vh')
- 90.0 0.0 -90.0: Euler rotation angles alpha, beta, gamma
- 4.e-3: Value of electron density (or LDOS) for isosurface
- -0.08 +0.01 +0.12: Saturation range for color function
- 'unformatted': Data format ('formatted'|'unformatted')
- '/CPS': Output option ('/XWIN'|'/PS'|'/CPS'|+more)

Viewpoint is always from above (positive z axis)

To view the system from a different angle, rotate it with the Euler angles
**Input of PLRHO: plrho.dat**

| 'h2o' | System label, used to name input files |
| 'vh'  | Function(s) to plot ("rho"|"ldos"|"spin"|"vt"|"vh") |
| 90.0 0.0 -90.0 | Euler rotation angles alpha, beta, gamma |
| 4.e-3 | Value of electron density (or LDOS) for isosurface |
| -0.08 +0.01 +0.12 | Saturation range for color function |
| 'unformatted' | Data format ('formatted'|'unformatted') |
| '/CPS' | Output option ('/XWIN'|'/PS'|'/CPS'|+more) |

**Example:** view from –y (Euler angles = 90 -90 -90)
**Input of PLRHO: plrho.dat**

<table>
<thead>
<tr>
<th>Key</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>'h2o'</td>
<td>System label, used to name input files</td>
</tr>
<tr>
<td>'vh'</td>
<td>Function(s) to plot ('rho'</td>
</tr>
<tr>
<td>90.0</td>
<td>Euler rotation angles alpha, beta, gamma</td>
</tr>
<tr>
<td>4.0e-3</td>
<td>Value of electron density (or LDOS) for isosurface</td>
</tr>
<tr>
<td>-0.08</td>
<td>Saturation range for color function</td>
</tr>
<tr>
<td>+0.01</td>
<td></td>
</tr>
<tr>
<td>+0.12</td>
<td></td>
</tr>
<tr>
<td>'unformatted'</td>
<td>Data format ('formatted'</td>
</tr>
<tr>
<td>'/CPS'</td>
<td>Output option ('/XWIN'</td>
</tr>
</tbody>
</table>

**Example:** view from –y (Euler angles = 90 -90 -90)

**Reference axes** | **System axes**
Input of PLRHO: plrho.dat

<table>
<thead>
<tr>
<th>'h2o'</th>
<th>System label, used to name input files</th>
</tr>
</thead>
<tbody>
<tr>
<td>'vh'</td>
<td>Function(s) to plot ('rho'</td>
</tr>
<tr>
<td>90.0</td>
<td>Euler rotation angles alpha, beta, gamma</td>
</tr>
<tr>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>-90.0</td>
<td></td>
</tr>
<tr>
<td>4.e-3</td>
<td>Value of electron density (or LDOS) for isosurface</td>
</tr>
<tr>
<td>-0.08</td>
<td></td>
</tr>
<tr>
<td>+0.01</td>
<td></td>
</tr>
<tr>
<td>+0.12</td>
<td></td>
</tr>
<tr>
<td>'unformatted'</td>
<td>Data format ('formatted'</td>
</tr>
<tr>
<td>'/CPS'</td>
<td>Output option ('/XWIN'</td>
</tr>
</tbody>
</table>

Example: view from –y (Euler angles = 90 -90 -90)

Reference axes  System axes

\[\text{Alpha: first rotation around } z\]
Input of PLRHO: plrho.dat

<table>
<thead>
<tr>
<th>Input</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>'h2o'</td>
<td>System label, used to name input files</td>
</tr>
<tr>
<td>'vh'</td>
<td>Function(s) to plot ('rho'</td>
</tr>
<tr>
<td>90.0 0.0 -90.0</td>
<td>Euler rotation angles alpha, beta, gamma</td>
</tr>
<tr>
<td>4.e-3</td>
<td>Value of electron density (or LDOS) for isosurface</td>
</tr>
<tr>
<td>-0.08 +0.01 +0.12</td>
<td>Saturation range for color function</td>
</tr>
<tr>
<td>'unformatted'</td>
<td>Data format ('formatted'</td>
</tr>
<tr>
<td>'/CPS'</td>
<td>Output option ('/XWIN'</td>
</tr>
</tbody>
</table>

Example: view from –y (Euler angles = 90 -90 -90)

Reference axes | System axes

Beta: rotation around y
Input of PLRHO: plrho.dat

| 'h2o' | System label, used to name input files |
| 'vh'  | Function(s) to plot ('rho'|'ldos'|'spin'|'vt'|'vh') |
| 90.0 0.0 -90.0 | Euler rotation angles alpha, beta, gamma |
| 4.e-3 | Value of electron density (or LDOS) for isosurface |
| -0.08 +0.01 +0.12 | Saturation range for color function |
| 'unformatted' | Data format ('formatted'|'unformatted') |
| '/CPS' | Output option ('/XWIN'|'/PS'|'/CPS'|+more) |

Example: view from –y (Euler angles = 90 -90 -90)

Reference axes  System axes

Gamma: second rotation around z
### Input of PLRHO: plrho.dat

<table>
<thead>
<tr>
<th>Key</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>'h2o'</td>
<td>System label, used to name input files</td>
</tr>
<tr>
<td>'vh'</td>
<td>Function(s) to plot ('rho'</td>
</tr>
<tr>
<td>90.0 0.0 -90.0</td>
<td>Euler rotation angles alpha, beta, gamma</td>
</tr>
<tr>
<td>.e-3</td>
<td>Value of electron density (or LDOS) for isosurface</td>
</tr>
<tr>
<td>-0.08 +0.01 +0.12</td>
<td>Saturation range for color function</td>
</tr>
<tr>
<td>'unformatted'</td>
<td>Data format ('formatted'</td>
</tr>
<tr>
<td>'/CPS'</td>
<td>Output option ('/XWIN'</td>
</tr>
</tbody>
</table>
**Input of PLRHO: plrho.dat**

<table>
<thead>
<tr>
<th>'h2o'</th>
<th>System label, used to name input files</th>
</tr>
</thead>
<tbody>
<tr>
<td>'vh' Function(s) to plot ('rho'</td>
<td>'ldos'</td>
</tr>
<tr>
<td>90.0 0.0 0.0 90.0 0.0</td>
<td>Euler rotation angles alpha, beta, gamma</td>
</tr>
<tr>
<td>4.e-3</td>
<td>Value of electron density (or LDOS) for isosurface</td>
</tr>
<tr>
<td>-0.08 +0.01 +0.12 Saturation range for color function</td>
<td></td>
</tr>
<tr>
<td>'unformatted' Data format ('formatted'</td>
<td>'unformatted')</td>
</tr>
<tr>
<td>'/CPS' Output option ('/XWIN'</td>
<td>'/PS'</td>
</tr>
</tbody>
</table>

- **First value:** minimum saturation range **red**
- **Second value:** mean saturation range **white**
- **Third value:** maximum saturation range **blue**

Value of colouring function

- **Pure red**
- **Interpolation red/white**
- **Interpolation white/blue**
- **Pure blue**
Output of PLRHO

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>'h2o'</td>
<td>System label, used to name input files</td>
</tr>
<tr>
<td>'vh'</td>
<td>Function(s) to plot ('rho'</td>
</tr>
<tr>
<td>90.0 0.0 -90.0</td>
<td>Euler rotation angles alpha, beta, gamma</td>
</tr>
<tr>
<td>4.e-3</td>
<td>Value of electron density (or LDOS) for isosurface</td>
</tr>
<tr>
<td>-0.08 +0.01 +0.12</td>
<td>Saturation range for color function</td>
</tr>
<tr>
<td>'unformatted'</td>
<td>Data format ('formatted'</td>
</tr>
<tr>
<td>'/CPS'</td>
<td>Output option ('/XWIN'</td>
</tr>
</tbody>
</table>

- screen
- grey-scale postscript
- colour postscript
Output of PLRHO

- 'h2o' System label, used to name input files
- 'vh' Function(s) to plot ('rho'|'ldos'|'spin'|'vt'|'vh')
- 90.0 0.0 -90.0 Euler rotation angles alpha, beta, gamma
- 4.e-3 Value of electron density (or LDOS) for isosurface
- -0.08 +0.01 +0.12 Saturation range for color function
- 'unformatted' Data format ('formatted'|'unformatted')
- '/CPS' Output option ('/XWIN'|'/PS'|'/CPS'|+more)

H₂O molecule
Summary of different tools for post-processing and visualization

DENCHAR

PLRHO

DOS, PDOS

MACROAVE

SIESTA’s output with XCRYSDEN
Density Of States (DOS)

\[ g(E)dE \] the number of one-electron levels between \( E \) and \( E + dE \).
Density Of States (DOS)

\( g(E) \, dE \) the number of one-electron levels between \( E \) and \( E + dE \)

\[
g(E) = \frac{1}{N_{\vec{k}}} \sum_{\text{bands}} \sum_{\vec{k}} \delta \left( E - E_{\vec{k}}(\vec{k}) \right)
\]

Units: (Energy\(^{-1}\))
Projected Density Of States (PDOS)

\[ g_\mu(E) \, dE \] the number of one-electron levels with weight on orbital \( \mu \) between \( E \) and \( E + dE \)

\[
g_\mu(E) = \frac{1}{N_k} \sum_i \sum_{\vec{k}} \sum_{\nu} c_{\nu i}^*(\vec{k}) c_{\mu i}(\vec{k}) S_{\nu \mu}(\vec{k}) \delta \left( E - E_i(\vec{k}) \right)
\]

- Coefficients of the eigenvector \( \psi_i(\vec{k}) \)
- Overlap matrix of the atomic basis with eigenvalue \( E_i(\vec{k}) \)

Units: (Energy)^{-1}

Relation between the DOS and PDOS:

\[
g(E) = \sum_\mu g_\mu(E)
\]
The eigenvalues are broadening by a gaussian to smooth the shape of the DOS and PDOS.

\[ \delta \left( E - E_i(\vec{k}) \right) \rightarrow \frac{1}{\sigma \sqrt{\pi}} e^{-\frac{(E - E_i(\vec{k}))^2}{\sigma^2}} \]
Two step procedure to produce smooth DOS and PDOS

First:

Run a simulation with a sensible (converged) number of k-points

kgrid_cutoff

%block kgrid_Monkhorst_Pack

Get converged geometry and density matrix

Second:

Starting from the previously converged geometry and density matrix, run a single SCF step with fixed geometry, with many more k-points

DM.UseSaveDM .true.
MaxSCFIterations 1
MD.NumCGsteps (or equivalent) 0
Increase number of k-points (see above)
%block ProjectedDensityOfStates
How to compute the DOS and PDOS

%block ProjectedDensityOfStates
   -20.0  10.0  0.200 500 eV
%endblock ProjectedDensityOfStates

-20.0 10.0 : Energy window where the DOS and PDOS will be computed
How to compute the DOS and PDOS

%block ProjectedDensityOfStates
   -20.0  10.0  0.200 500 eV
%endblock ProjectedDensityOfStates

-20.0  10.0 : Energy window where the DOS and PDOS will be computed

0.200 : Peak width of the gaussian used to broad the eigenvalues (energy)
How to compute the DOS and PDOS

%block ProjectedDensityOfStates
  -20.0  10.0  0.200  500  eV
%endblock ProjectedDensityOfStates

-20.0  10.0 : Energy window where the DOS and PDOS will be computed

0.200 : Peak width of the gaussian used to broad the eigenvalues (energy)

500 : Number of points in the histogram
How to compute the DOS and PDOS

%block ProjectedDensityOfStates
   -20.0  10.0  0.200 500 eV
%endblock ProjectedDensityOfStates

-20.0  10.0 : Energy window where the DOS and PDOS will be computed

0.200 : Peak width of the gaussian used to broad the eigenvalues (energy)

500 : Number of points in the histogram

eV : Units in which the previous energies are introduced
Output for the Density Of States
SystemLabel.DOS

<table>
<thead>
<tr>
<th>Energy (eV)</th>
<th>DOS Spin Up (eV(^{-1}))</th>
<th>DOS Spin Down (eV(^{-1}))</th>
</tr>
</thead>
<tbody>
<tr>
<td>-15.99598</td>
<td>0.03075</td>
<td>0.00000</td>
</tr>
<tr>
<td>-15.97596</td>
<td>0.03580</td>
<td>0.00000</td>
</tr>
<tr>
<td>-15.95594</td>
<td>0.04115</td>
<td>0.00000</td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
</tbody>
</table>
Output for the Projected Density Of States

SystemLabel.PDOS

```xml
<pdos>
  <nspin>1</nspin>
  <orbitals> 26</orbitals>
  <energy_values units="eV">
    -19.99998
    -19.97996
    -19.95994
    .
    .
  </energy_values>
  <orbital
    index="1"
    atom_index="1"
    species="Si"
    position=" 0.000000 0.000000 0.000000"
    n="3"
    l=" 0"
    m=" 0"
    z="1"
  >
  <data>
    0.00000
    0.00000
    0.00000
    .
    .
    .
  </data>
</orbital>
</pdos>
```

Written in XML

Energy Window

One element <orbital> for every atomic orbital in the basis set
How to digest the SystemLabel.PDOS file

During the **compilation** of SIESTA

For some compilers, the libwxml.a library needs to be compiled with 
“-DWXML_INIT_FIX”
(see known issues in http://fisica.ehu.es/ag/siesta-extra/issues.html)

**pdosxml** (by Alberto García)

Siesta/Util/pdosxml

Edit the readme file to:

- Learn how to select the orbitals whose PDOS will be accumulated
- How to compile the code
- How to run the code

**fmpdos** (by Andrei Postnikov)

Download from

[http://www.home.uni-osnabrueck.de/apostnik/download.html](http://www.home.uni-osnabrueck.de/apostnik/download.html)

Compile and follow the instructions at run-time
Normalization of the DOS and PDOS

\[ \int_{-\infty}^{+\infty} g(E) \, dE = \text{Number of bands per k-point} = \text{Number of atomic orbitals in the unit cell} \]

\[ \int_{-\infty}^{+\infty} g(E) n(E) \, dE = \text{Number of electrons in the unit cell} \]

Occupation factor at energy \( E \)
Example of DOS and PDOS

\[
g_\mu(E) = \frac{1}{N_k} \sum_i \sum_{\vec{k}} \sum_{\nu} c_{\nu i}(\vec{k})^* c_{\mu i}(\vec{k}) S_{\nu \mu}(\vec{k}) \delta \left( E - E_i(\vec{k}) \right)
\]


SIESTA, single-zeta polarized basis


All electron calculation
Summary of different tools for post-processing and visualization

DENCHAR

PLRHO

DOS, PDOS

MACROAVE

SIESTA’s output with XCRYSDEN
How to extract from the immense detail provided by first-principles calculations on surfaces reliable values of the physical quantities of interest
Physical quantities of interest in surfaces and interfaces

- Charge densities at the surface/interface
- Dipole moment densities at the surface/interface
- Work functions (surfaces) and band offsets (interfaces)

\[ V_{BO} = \Delta E_y + \Delta V \]

- \( \Delta E_y \): the band structure term
- Difference of the top of the valence bands
- From two independent bulk band structure calculations of the bulk material

- \( \Delta V \): jump of the average electrostatic potential
- Contains all the intrinsic interface effects
- Obtained by nanosmoothing the electrostatic potential
First step: average in the plane

\[ \overline{V}(z) = \frac{1}{S} \int_S dx dy \, V(\vec{r}) \]
Second step: nanosmooth the planar average on the $z$-direction

$$
\bar{V}(z) = \int dz' \ f(z - z')\bar{V}(z')
$$
Atomic scale fluctuations are washed out by filtering the magnitudes via convolution with smooth functions

$$ \overline{\rho}(z) = \int dz' f(z - z') \overline{\rho}(z') $$

$$ f(z - z') = \int dz'' \omega_{l_1}(z - z'') \omega_{l_2}(z'' - z') $$
$\Delta V$ is readily obtained from the nanosmoothed potential

![Diagram showing vacuum, BaTiO$_3$, and vacuum with a graph depicting the electrostatic potential (eV).]
How to compile MACROAVE...

$ cd ~/siesta/Util/Macroave/Src/
$ make
ifort -c -03 defs_basis.f90
ifort -c -03 defs_common.f90
ifort -c -03 io.f
ifort -c -03 iorho.f
ifort -c -03 paste.f
ifort -c -03 macroave.f
ifort -c -03 thetaft.f
ifort -c -03 surpla.f
ifort -c -03 volcel.f
ifort -c -03 recipes.f
ifort -c -03 hdr_io.f90
ifort -o macroave.x \
   defs_basis.o defs_common.o io.o iorho.o paste.o macroave.o thetaft.o
   surpla.o volcel.o recipes.o hdr_io.o

Use the **same arch.make** file as for the compilation of **serial SIESTA**

...and where to find the User’s Guide and some Examples

$ cd ~/siesta/Util/Macroave
$ ls
Docs Examples Src
How to run MACROAVE

SIESTA

SaveRho .true.
SaveTotalCharge .true.
SaveIonicCharge .true.
SaveDeltaRho .true.
SaveElectrostaticPotential .true.
SaveTotalPotential .true.

Output of SIESTA required by MACROAVE

SystemLabel.RHO
SystemLabel.TOCH
SystemLabel.IOCH
SystemLabel.DRHO
SystemLabel.VH
SystemLabel.VT

MACROAVE

Prepare the input file macroave.in
$ ~/siesta/Util/Macroave/Src/macroave.x

Depending on what you want to nanosmooth

You do not need to rerun SIESTA to run MACROAVE as many times as you want
# The same code with the same input runs with information provided by

<table>
<thead>
<tr>
<th>Siesta</th>
<th>which code have you used to get the input data?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Potential</td>
<td>Quantity that will be nanosmoothed</td>
</tr>
<tr>
<td>Interface</td>
<td>Name of the file where the input data is stored</td>
</tr>
<tr>
<td>2</td>
<td>Number of convolutions required to calculate the macro. ave.</td>
</tr>
<tr>
<td>7.47622255</td>
<td>First length for the filter function in macroscopic average</td>
</tr>
<tr>
<td>7.56114385</td>
<td>Second length for the filter function in macroscopic average</td>
</tr>
<tr>
<td>500</td>
<td>Total number of electrons</td>
</tr>
<tr>
<td>spline</td>
<td>Type of interpolation</td>
</tr>
</tbody>
</table>

(indeed it should be quite straightforward to generalize to any other code)
## Input of MACROAVE: macroave.in

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Siesta</td>
<td># Which code have you used to get the input data?</td>
</tr>
<tr>
<td>Potential</td>
<td># Quantity that will be nanosmoothed</td>
</tr>
<tr>
<td>Interface</td>
<td># Name of the file where the input data is stored</td>
</tr>
<tr>
<td>2</td>
<td># Number of convolutions required to calculate the macro. ave.</td>
</tr>
<tr>
<td>7.47622255</td>
<td># First length for the filter function in macroscopic average</td>
</tr>
<tr>
<td>7.56114385</td>
<td># Second length for the filter function in macroscopic average</td>
</tr>
<tr>
<td>500</td>
<td># Total number of electrons</td>
</tr>
<tr>
<td>spline</td>
<td># Type of interpolation</td>
</tr>
</tbody>
</table>

### Name of the magnitude that will be nanosmoothed

- **Potential:** SystemLabel.VH
- **Charge:** SystemLabel.RHO
- **TotalCharge:** SystemLabel.TOCH
**Input of MACROAVE: macroave.in**

<table>
<thead>
<tr>
<th>Code</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Siesta</td>
<td>Which code have you used to get the input data?</td>
</tr>
<tr>
<td>Potential</td>
<td>Quantity that will be nanosmoothed</td>
</tr>
<tr>
<td>Interface</td>
<td>Name of the file where the input data is stored</td>
</tr>
<tr>
<td>2</td>
<td>Number of convolutions required to calculate the macro. ave.</td>
</tr>
<tr>
<td>7.47622255</td>
<td>First length for the filter function in macroscopic average</td>
</tr>
<tr>
<td>7.56114385</td>
<td>Second length for the filter function in macroscopic average</td>
</tr>
<tr>
<td>500</td>
<td>Total number of electrons</td>
</tr>
<tr>
<td>spline</td>
<td>Type of interpolation</td>
</tr>
</tbody>
</table>
**Input of MACROAVE: macroave.in**

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Siesta</td>
<td>Which code have you used to get the input data?</td>
</tr>
<tr>
<td>Potential</td>
<td>Quantity that will be nanosmoothed</td>
</tr>
<tr>
<td>Interface</td>
<td>Name of the file where the input data is stored</td>
</tr>
<tr>
<td>2</td>
<td>Number of convolutions required to calculate the macro. ave.</td>
</tr>
<tr>
<td>7.47622255</td>
<td>First length for the filter function in macroscopic average</td>
</tr>
<tr>
<td>7.56114385</td>
<td>Second length for the filter function in macroscopic average</td>
</tr>
<tr>
<td>500</td>
<td>Total number of electrons</td>
</tr>
<tr>
<td>spline</td>
<td>Type of interpolation</td>
</tr>
</tbody>
</table>

**Number of square filter functions used for nanosmoothing**

1. Surfaces
2. Interfaces and superlattices
Input of MACROAVE: macroave.in

Siesta # Which code have you used to get the input data?
Potential # Quantity that will be nanosmoothed
Interface # Name of the file where the input data is stored
2 # Number of convolutions required to calculate the macro. ave.
7.47622255 # First length for the filter function in macroscopic average
7.56114385 # Second length for the filter function in macroscopic average
500 # Total number of electrons
spline # Type of interpolation

Length of the different square filter functions (in bohrs)
### Input of MACROAVE: macroave.in

<table>
<thead>
<tr>
<th>Siesta</th>
<th>Which code have you used to get the input data?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Potential</td>
<td>Quantity that will be nanosmoothed</td>
</tr>
<tr>
<td>Interface</td>
<td>Name of the file where the input data is stored</td>
</tr>
<tr>
<td>2</td>
<td>Number of convolutions required to calculate the macro. ave.</td>
</tr>
<tr>
<td>7.47622255</td>
<td>First length for the filter function in macroscopic average</td>
</tr>
<tr>
<td>7.56114385</td>
<td>Second length for the filter function in macroscopic average</td>
</tr>
<tr>
<td>500</td>
<td>Total number of electrons</td>
</tr>
<tr>
<td>spline</td>
<td>Type of interpolation</td>
</tr>
</tbody>
</table>

**Total number of electrons**

(used only to renormalize if we nanosmooth the electronic charge)
## Input of MACROAVE: macroave.in

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Siesta</td>
<td>Which code have you used to get the input data?</td>
</tr>
<tr>
<td>Potential</td>
<td>Quantity that will be nanosmoothed</td>
</tr>
<tr>
<td>Interface</td>
<td>Name of the file where the input data is stored</td>
</tr>
<tr>
<td>2</td>
<td>Number of convolutions required to calculate the macro. ave.</td>
</tr>
<tr>
<td>7.47622255</td>
<td>First length for the filter function in macroscopic average</td>
</tr>
<tr>
<td>7.56114385</td>
<td>Second length for the filter function in macroscopic average</td>
</tr>
<tr>
<td>500</td>
<td>Total number of electrons</td>
</tr>
<tr>
<td>spline</td>
<td>Type of interpolation</td>
</tr>
</tbody>
</table>

### Type of interpolation from the SIESTA mesh to a FFT mesh

Spline or Linear
Output of MACROAVE

Planar average

\[ \bar{\rho}(z) = \frac{1}{S} \int_{S} dxdy \ \rho(\vec{r}) \]

Nanosmoothed

\[ \bar{\rho}(z) = \int dz' f(z - z') \bar{\rho}(z') \]

SystemLabel.PAV

SystemLabel.MAV

Format

<table>
<thead>
<tr>
<th>( z )</th>
<th>average</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0000000000000000</td>
<td>0.0000000000000000</td>
</tr>
<tr>
<td>0.000095367432</td>
<td>0.0000000000000000</td>
</tr>
<tr>
<td>0.000190734863</td>
<td>0.0000000000000000</td>
</tr>
<tr>
<td>\ldots</td>
<td>\ldots</td>
</tr>
</tbody>
</table>

Units

Coordinates: bohr  Potential: eV  Charge density: electrons/bohr\(^3\)
To learn more on nanosmoothing and how to compute work functions and band offsets with SIESTA

IOP PUBLISHING

TOPICAL REVIEW

Nanoscale smoothing and the analysis of interfacial charge and dipolar densities

Javier Junquera\textsuperscript{1,2}, Morrel H Cohen\textsuperscript{2,3} and Karin M Rabe\textsuperscript{2}

PHYSICAL REVIEW B 67, 155327 (2003)

First-principles calculation of the band offset at BaO/BaTiO\textsubscript{3} and SrO/SrTiO\textsubscript{3} interfaces

Javier Junquera,\textsuperscript{1} Magali Zimmer,\textsuperscript{1} Pablo Ordejón,\textsuperscript{2} and Philippe Ghosez\textsuperscript{1}