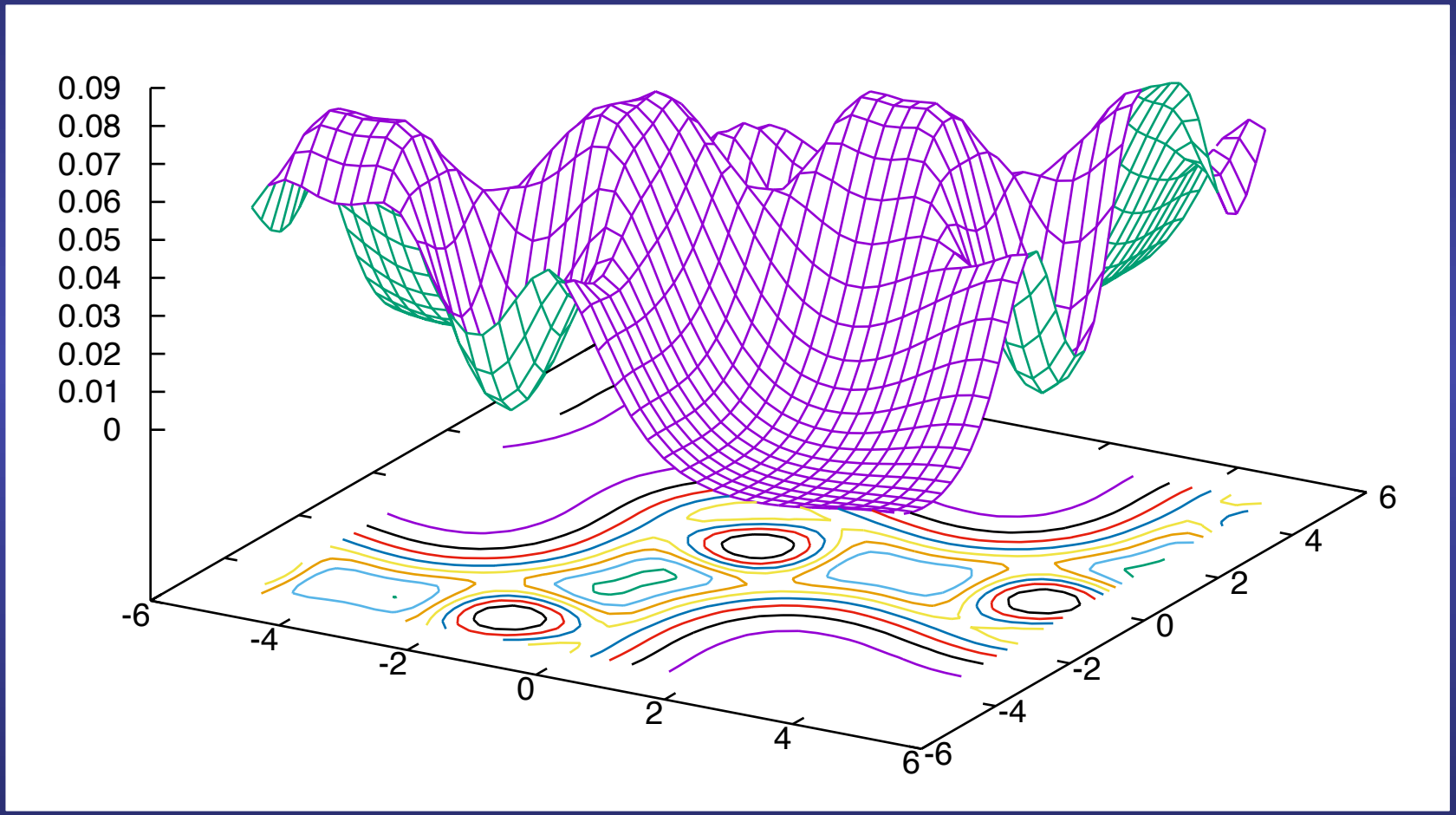
Plotting the charge density of bulk Si



Bulk Si: a covalent solid that crystallizes in the diamond structure

Go to the directory where the exercise on the structure of Si is stored

Inspect the input file, Si.fdf

More information at the Siesta web page http://www.icmab.es/siesta and follow the link Documentations, Manual

```
SystemName      Bulk Silicon
SystemLabel     Si
NumberOfSpecies         1
NumberOfAtoms           2
%block ChemicalSpeciesLabel
  1  14  Si
%endblock ChemicalSpeciesLabel

LatticeConstant      5.41119733025 Ang
%block LatticeVectors
  0.00  0.50  0.50
  0.50  0.00  0.50
  0.50  0.50  0.00
%endblock LatticeVectors

AtomicCoordinatesFormat          ScaledByLatticeVectors
%block AtomicCoordinatesAndAtomicSpecies
  0.00  0.00  0.00        1
  0.25  0.25  0.25        1
%endblock AtomicCoordinatesAndAtomicSpecies

%block kgrid_Monkhorst_Pack
   4  0  0  0.5
   0  4  0  0.5
   0  0  4  0.5
%endblock kgrid_Monkhorst_Pack
```

The theoretical lattice constant of Si for this first example

Diamond structure:

FCC lattice

+ a basis of two atoms

Sampling in k in the first Brillouin zone to achieve self-consistency

Bulk Si: a covalent solid that crystallizes in the diamond structure

Inspect the input file, Si.fdf

Take a look to these input variables to produce the required files to plot charge densities

More information at the Siesta web page
http://www.icmab.es/siesta and follow
the link Documentations, Manual

```
WriteDenchar            .true.        # Write information for DENCHAR
WriteWaveFunctions      .true.        # If .true. it writes to the output file
                                      #    a list of the wavefunctions actually
                                      #    written to the
                                      #    Systemlabel.selected.WFSX file
                                      #Specifies the k-points at which
                                      #   the electronic wavefunction
%block WaveFuncKPoints                #   coefficients are written
   0.000  0.000  0.000  from 1 to 10  # Gamma wavefuncs 1 to 10
%endblock WaveFuncKPoints             # The index of a wavefunction is defined
                                      #    by its energy, so that the first one
                                      #    has lowest energy.


# The user can also narrow the energy-range used with the
#    WFS.Energy.Min
#       and
#    WFS.Energy.Max
#       options (both are physical quantities that require a number
#                and energy unit)
```

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 run DENCHAR…

SIESTA

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

Only if you want to plot wave functions

Output of SIESTA required by DENCHAR

SystemLabel.PLD
SystemLabel.DIM
SystemLabel.DM
SystemLabel.selected.WFSX (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

How to compile DENCHAR…

Go to the directory with the package

$cd Util/Denchar/Src

And type

$make OBJDIR=Obj/

Where OBJDIR should point to the directory where the arch.make you want to use is located

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.WFSX must be present

Either one or the other (or both of them) must be .true.

Input of DENCHAR
How to specify the plane

{ Plane of the plot in 2D mode
{ x-y plane in 3D mode

- **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

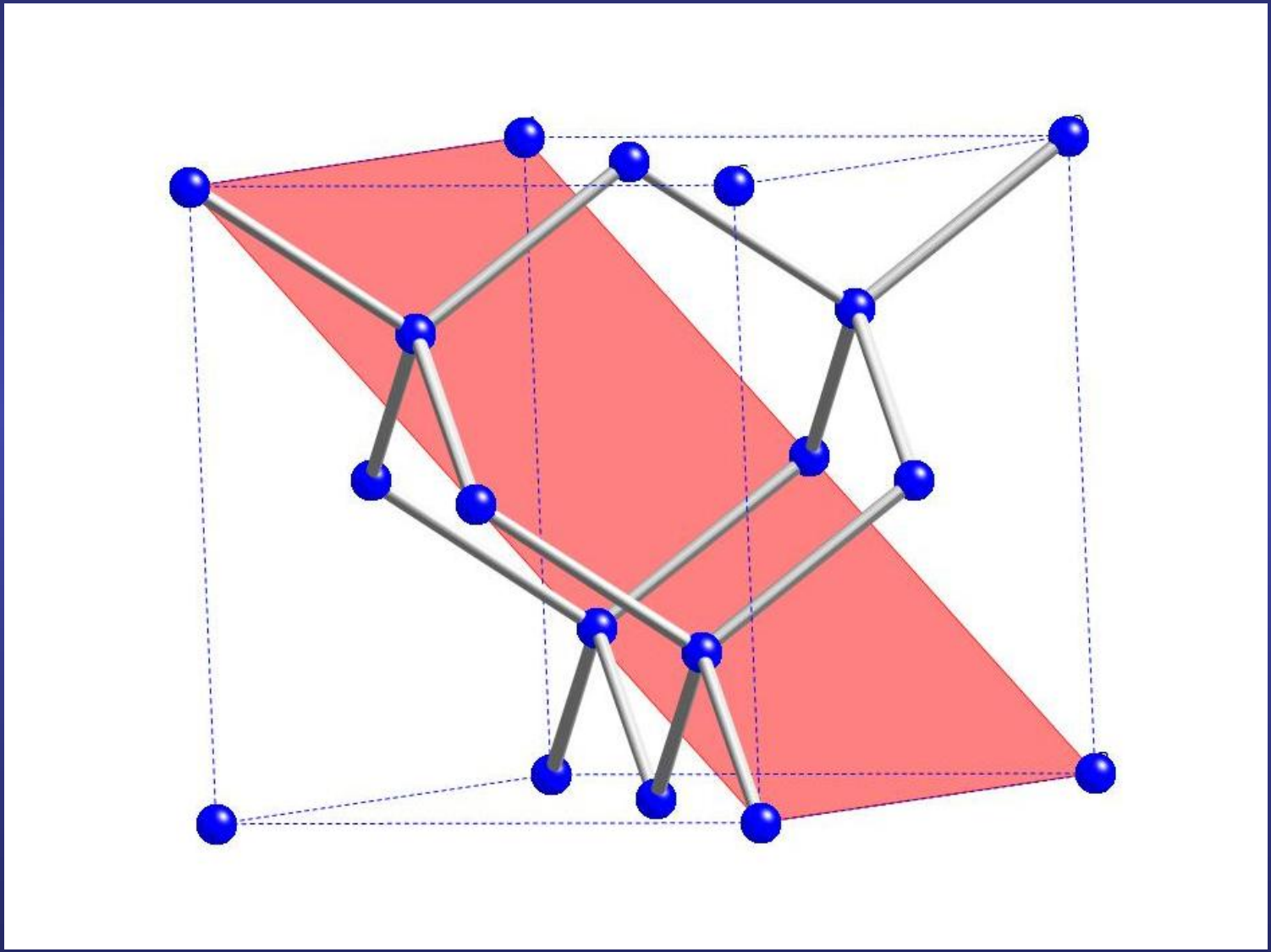
number of points in the grid

- **Different variables described in the User Guide (take a look to the Examples)**

Input of DENCHAR

```
#  *********************************************************************
#                      Input variables for Denchar
#  (besides SystemLabel, NumberOfSpecies and ChemicalSpecies, defined above)
#  *********************************************************************

Denchar.TypeOfRun           2D

Denchar.PlotCharge          .true.
Denchar.PlotWaveFunctions   .true.

Denchar.CoorUnits      bohr            # Format for coordinate of the points
                                       # Bohr
                                       # Ang

Denchar.DensityUnits   Ele/bohr**3     # Units of Charge Density
                                       # Ele/bohr**3
                                       # Ele/Ang**3
                                       # Ele/UnitCell

Denchar.MinX           -5.0  bohr      # Minimum coord of the window in X-dir
Denchar.MaxX            5.0  bohr      # Maximum coord of the window in X-dir

Denchar.MinY           -5.0  bohr      # Minimum coord of the window in Y-dir
Denchar.MaxY            5.0  bohr      # Maximum coord of the window in Y-dir

Denchar.NumberPointsX    40            # Number of points in X-axis
Denchar.NumberPointsY    40            # Number of points in Y-axis

Denchar.PlaneGeneration  ThreeAtomicIndices      # Option to generate the plane
                                                 # NormalVector
                                                 # TwoLine
                                                 # ThreePoints
                                                 # ThreeAtomicIndices
%block Denchar.Indices3Atoms                # Indices of three atoms
   1  2  3
%endblock Denchar.Indices3Atoms
```

Chosen plane in the example



To produce the figures of the densities and wave functions in real space

$ siesta < Si.fdf > Si.out

The selected wave functions are written in a file called
Si.selected.WFSX

and the files required to run DENCHAR are
Si.PLD
Si.DIM
Si.DM
ChemicalSpecies.ion (one for each chemical species)

To run denchar and produce the corresponding output files for the wavefunctions, we have to
rename the SystemLabel.selected.WFSX to SystemLabel.WFSX

$ cp Si.selected.WFSX Si.WFSX

run DENCHAR
$ denchar < Si.fdf

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.K#.WF#.REAL

.CON.K#.WF#.IMAG

.CON.K#.WF#.MOD

.CON.K#.WF#.PHASE

where # after K is the number of k-point in the list, and # after the WF is the number of wavefunction for that k-point (in order of energy).

The suffix (REAL, IMAG, MOD, PHASE) is self-explanatory

(If spin polarized, suffix .UP or .DOWN)

Format

x_i	y_i	$f(x_i, y_i)$
-5.00000	-5.00000	-0.00560
-5.00000	-4.74359	-0.00402
.	.	.
.	.	.

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)

- same format as before but with the suffix .cube

Format

Gaussian Cube format

- Atomic coordinates and grid points in the reference frame given in the input
- Reference frame orthogonal

Visualization of the charge density

If you have python with the libraries numpy and gnuplot installed

```
$ surf.py Si.CON.SCF
$ 2dplot.py Si.CON.SCF
```

Replace the name of the file for one of your choice

If not, you can edit the file surf.gplot

```
$ vi surf.gplot
```

```
set parametric
set style data lines
set hidden
set contour base
set cntrparam levels auto 10
splot "Si.CON.SCF" using 1:2:3 with lines notitle
```

change the name of the file you want to plot in the last line, save the file and run:

```
$ gnuplot surf.gplot
```

Visualization of the charge density