Contents

1 Introduction

This program solves the Kohn-Sham equations of the density-functional theory [1] in axial symmetry. In this document the symmetry axis is called the z-axis and the radial axis r. In this geometry the Schrdinger equation is separable, and the wave-functions can be written as $\psi^s_{m\mathbf{k}n}(r,z,\phi) = e^{im\phi}U^s_{m\mathbf{k}n}(r,z)$.

The physical systems that can be handled by the program include arbitrary combinations of atoms on the z-axis and axially symmetric pieces of (stabilized[2], normal[3] or ultimate [4]) jellium. In practice, we have implemented jellium cones, spheres, cylinders, infinite wires (with periodic boundary conditions in the z-direction) and infinite planar slabs (requiring “periodic” boundary conditions in the r-direction) and a possibility to smoothly join a uniform stabilized jellium wire with a freely deformable “neck”-part consisting of ultimate jellium.

2 The input file

All the commands of the calculation are written in one input file. The file name doesn't have to be INPUT, even if it is so called in this document. To tell the program how to find the input file it's name have to be written in the file named CYL2_ARG. This file has only one word inside: the name of the input file.

In this document we introduce some basic commands, and go through a couple of simple example calculations. Using these commands it is possible to calculate some basic electronic structure calculations. The syntax of the input file is case insensitive; both lower and uppercase commands are accepted, you can even mix lowercase and uppercase in the same command. However, in this document commands are written in capital letters. Every command is written in separate line and empty lines are not acceptable. You can add comments in the INPUT file. The comment lines must begin with a per-cent sign (%) followed by a space.

The input file starts with the command:

SET MODE START

or

SET MODE CONTINUE

depending on whether we are starting a new calculation or continuing an old one. After that, we give values to certain “environment” variables using the “SET”-command. Most environment variables have default values, and it is not necessary to mention these variables in the input-file, unless you need to change their values. However, the following variables have to be explicitly given in the input file: MAX_M (highest value for the azimuthal quantum number m) M_SUBBANDS (list of MAX_M+1 integers giving the number of states to be calculated for each m-value starting from 0) GRID, CELL, and K_POINTS. Once these settings are given, the program allocates the wave-functions and potentials. There are some other parameters (at least ASA), that affect the number of the wavefunctions to be allocated. These have to be given at the beginning of the input file, so that memory allocation goes properly. (If you try to change these settings “too late”, the program gives an error message.

After parameter settings the calculation is started with the command

RUN

After iterations it is possible to do more parameters changes and start running again. This is usually done when calculation is started: at the begin one scf-iteration is done in full multigrid mode and after it the mode is changed to V-cycles and the iterations are started again.

After the iterations you can add commands for postprocessing the results. Such postprocessing includes the possibility of plotting the individual Kohn-Sham orbitals (saving them to text files that can be accessed by matlab or other programs), or calculation of the local density of states. The last command in the input file should be

QUIT

An empty line in input file works (at the time of writing) also as a quit command.

3 Number of grid points and the cell size

The grid size is set using command:

SET GRID r z

where r is the number of grid points in r-direction and z is in z-direction. In order to get multigrid algorithm work properly the grid size have be of the form

$C * 2^m - 1$ (Dirichlet or mirror boundary conditions)
$C * 2^m$ (Periodic boundary conditions)

C and m are integers. They don't have to be same in r and z direction. m should be biker than 3. Five is quite good.

The size of the calculation cell is set using command:

SET CELL r z

where r is the length in atomic units in r-direction and z is in z-direction. The labeling of the z-axis goes from -z/2 to z/2.

4 The potentials in the program

4.1 The sign of the charge

The sign of the charge is set so that the positive charge is negative and the negative is positive - in density-functional theory it is natural that the electron density is a positive quantity.

4.2 The pseudopotentials

The pseudopotentials are stored in separate files. These files have the same format as those used by the program `MIKA/rspace`. The atoms are put in their places using the command

ATOM pot_filename z

where z is the z-coordinate of the atom. Because of the cylinder symmetry, atoms can locate only on the z-axis, so r-axis is not needed. The program reads the file with the pathname `BASE_DIR/potentials/pot_filename`. The BASE_DIR environment variable defaults to " /MIKA". It can be changed in the input file for example like this : SET BASE_DIR "/wrk/username/MIKA".

4.3 Jellium parts of the system

The jellium is constant positive background charge. The density of positive charge is described using r_s parameter. It is the radius of the ball, which has one basic charge, e inside of it. The jellium piece have to keep inside an integer number of basic charges. So the size of the jellium piece can't be arbitrary.

In the program there the jellium is created using:

WIRE $r\ r_s\ h_1\ h_2$

Which creates jellium with Dirichlet boundary conditions in r-direction. r is the radius and $h_1\ h_2$, the beginning and the end of the wire ($h_1 < h_2$)

SLAB $r_s\ h_1\ h_2$

Creates the jellium with periodic boundary conditions in the r-direction. The radius is not needed, because the jellium goes to the boundary of the calculation cell. To calculate slabs properly, you should also have the line "SET ASA .true." This setting ensures that the boundary conditions at the large-r boundary are selected properly. It also automatically selects a two k-point sampling of the Brillouin zone of the area-covering lattice of circles: The gamma-point and the k-point at the Brillouin-zone boundary are used. These correspond to zero derivative and zero value at large r, respectively. (Or actually to EVEN and ODD mirror-type boundary conditions in the implementation).

CONE $r_s\ r_1\ r_2\ h_1\ h_2$

Creates a jellium cone. r_1 and r_2 are the radius of the cone at the lower and upper edge end of the cone, respectively.

The stabilized jellium is possible to get using command

SET STABILIZED_JELLIUM .TRUE.

otherwise the normal jellium is used. Similarly,

SET ULTIMATE_JELLIUM .TRUE.

ensures, that the positive background charge at each grid point is exactly the same as the electron density. The system is thus completely neutral, and free to deform to a energy-minimizing shape (with the constraint of axial symmetry). Since completely free shapes are not very interesting, you can "freeze" the potential at some part of the system using the command

FIXED_WIRE_V fixed_V_file Z_min Z_max .

The program then reads from disk a file called `fixed_V_file` containing the reference potential (for example that of a uniform jellium wire), and forces the potential to this value in the region with $Z_min < z < Z_max$.

5 Indexes of the Kohn-Sham states

5.1 m and n

The maximum value of the m quantum number is given using the command

SET MAX_M m_max.

After that one has to give the number n of states for every value of m (m=0,1,...,8):

SET M_SUBBANDS 23 20 14 10 8 5 4 3 2

For technical reasons, using n=1 may (at the time of writing) lead to problems – use $n > 2$ for all m. Each state with $m > 0$ is twofold degenerate.

5.2 k

The number of k_z-points is set using command

`SET KZ_POINTS` k_z^{max}

For backwards compatibility, `K_POINTS` is equivalent to `KZ_POINTS`. The actual number of k$_z$ points use to sample the 1D-Brillouin zone for systems periodic in z-direction is `K_POINTS`+1 – thus, for nonperiodic systems one should have `SET K_POINTS 0`. The k-point sampling used in the case $k_z^{max} > 0$ includes the points

$$k_z = \frac{i}{k_z^{max}} \frac{\pi}{L_z}, \quad i = 1, \ldots, k_z^{max}.$$

Because of inversion symmetry, the points with negative k_z yield the same density. Thus, when constructing the density, the points $k_z = 0$ (the Γ-point) and $k_z = \frac{\pi}{L_z}$ (the L-point) have only half the weight of the other k_z-points.

When using the Γ or (Γ + L) sampling (`KZ_POINTS` 0 or 1) it is worth while to save some computation time and use real wave-functions instead of complex ones. This is possible to do using command

SET COMPLEX_NUMBERS .TRUE./.FALSE.

The lateral k-points are handled differently. As mentioned before, in the case of a slab calculation, one has to "SET ASA .true." which ensures the use of two lateral k-points. Both of these can be handled with real wave-functions. The command

SET ASA_PERIODIC .TRUE.

is a shortcut for the settings necessary to handle a system with periodic boundary conditions in both z- and r-directions

5.3 s

s is the spin quantum number and it is in this program 1 or 2. The calculation can be spin compensated or not. The choice is done using command

SET SPIN_COMPENSATED .TRUE./.FALSE..

5.4 p

Inserting

SET MIRROR_PLANE_DIRICHLET .true.

in the input file is a shortcut for the settings necessary for a finite system with a horizontal symmetry plane. Then each wave-function has a well defined parity p with respect to that plane.

6 Solving algorithms

It is possible to chose multigrid or conjugate gradients for solving algorithm. The chose is done using command

SET SCHRODINGER_SOLVER multigrid

or

SET SCHRODINGER_SOLVER conjugate_gradients

The multigrid solver is faster, and it is better to use it. Also the conjugate gradient method is currently not maintained, and the author does not guarantee that is works at all. In the multigrid solver there are two modes, which are used:

SET multigrid_mode full_multigrid

or

SET multigrid_mode V_cycle

The full multigrid is a good method to obtain an initial guess for the wave-functions at the begin of the calculation, while V_cycles should be used for the iterations toward self-consistency.

7 Plotting the wave functions

One can save the squared moduli $|U_{m\mathbf{k}n}(r,z)|^2$ of the wave functions to text files using command **plot**. The name of the file is of the form: wave.m=1n=2k=3s=1, where it m, n, k and s are the indices as described above. The k-index is internally a bit overloaded: it handles the physical k_z, $k_{lateral}$ and p indices.

The wavefunctions are mirrored with respect to the z-axis before writing. This produces nicer plots in matlab or other plotting program.

The command

PLOT m n k s

Plots single state which have quantum numbers m, n, k and s. k state includes both lateral and k_z quantum numbers (and p):

$$k = k_{lateral} * (\max(k_{lateral})) + k_z$$

PLOT m k s

Plot all the states which has these quantum states.

PLOT k s

Plots also all the states which has these quantum states.

PLOT

Plots all of the states.

The plot order is possible to give at the end of the INPUT-file, when the iterations are finished.

8 The continue of the calculation

It is possible to continue the old calculation. For doing this you should have INPUT, V_up, V_down, eig_in and the wave functions file from the previous calculation.

The wave functions are saved on the disk during the calculation using the command

SET SAVE_WF .TRUE.

The saving is done once every 10 minutes. So that a calculation is possible to cancel during the running, do some changes to the parameters and go on again. The wave functions are saved to the file called PSEUDO. It is possible to change this name using the command

SET WF_FILENAME *filename*

The continue of the calculation starts with changing the calculation mode in the input file to be

SET MODE CONTINUE

If the number of processors is different in old calculations than a new starting one, then the number of processors in old calculation have to give to the program using command

SET OLD_N_OF_PES n

where n is the number of processors. Then the program can read the wave functions correctly inside from the disk. The wave functions are read from the disk using the command

SET LOAD_WF .TRUE.

and the occupations of the electrons

SET GIVEN_OCCUPARION .TRUE.

Now it is possible to go on with the calculation and do some more self consistent iterations. It is also possible to carry on the calculation without any new iteration. This is the case if it is just wanted to plot some more out. In this case there have to be command

INIT

in same place than in usual it is RUN. The INIT do same initialization and allocations than RUN, and without it plotting is not working properly.

9 Boundary conditions

The boundary conditions are maybe easiest to explain in the context of their actual implementation. Since we are using finite differences to solve the partial differential equations, we need the values of the functions at *ghost*-points, slightly outside the calculation box. Below, the implemented boundary conditions are explained. The term *mirror point* refers to the mirror image of the ghost point with respect to the boundary, and the *periodic point* is the periodic image of the ghost point, that is within the calculation volume, and *boundary point* is the interior point closest to the ghost point.

DIRICHLET The value at the ghost point is not changed.
EVEN The value at ghost point is same as at the mirror point.
ODD The value at ghost point is the negative of that at the mirror point.
PERIODIC The value at ghost point is same as at the periodic point.
NPERIODIC The value at ghost point is the negative of that at the periodic point.
BLOCH The value at ghost point is that at the periodic point multiplied by the bloch factor $\exp(ik_zL_z)$ (only allowed in z-direction)
MIRROR Wildcard that will be assigned later EVEN or ODD according to $k_lateral$, for example.
CONSTANT The value at the ghost point is the average of values at all boundary points of the edge

The boundary conditions at each four edges of the calculation box in (r,z) space can be different. Furthermore, they can be different in the cases of the Poisson and the Schrödinger equations. However, the boundary condition at the axis is dictated by physics - in the case of the poisson equation it must be **EVEN**, whereas in the case of the Schrödinger equation it is **EVEN** for even m, and **ODD** for odd m. Therefore, you only give three arguments below

SET SCHRODINGER_BC DIRICHLET MIRROR DIRICHLET

The first argument being the boundary condition at the large r boundary, the second one at lower z-boundary and the third one at upper z-boundary. However, the user should not usually need to give these by hand : there are shortcuts, explained below, that also make sure all other settings are correct.

SET BOUNDARY_CONDITIONS ZERO

Is suitable for finite cluster calculations.

SET BOUNDARY_CONDITIONS PERIODIC

Means a periodic "wire" geometry - periodicity in the z-direction (implying **BLOCH** boundary conditions for the wave-functions, but zero dirichlet boundary conditions in the radial direction.

SET BOUNDARY_CONDITIONS ASA

This setting is used for structures periodic in the xy-plane, such as infinite jellium slab. To handle the jellium slabs, we adopt a method similar to the

Wigner-Seitz method[5] - we imagine the plane being filled by hexagons, and then replace these hexagons by area-covering circles. In order to sample the Brillouin zone of the lattice of circles we use two **k**-points, $\mathbf{k} = 0$ and $\mathbf{k}$ at the Brillouin zone boundary. The wavefunctions with $\mathbf{k} = 0$ are required to have a vanishing radial derivative at the radius of the circle whereas the wavefunctions with **k** at the Brillouin zone boundary vanish there. According to our calculations the model gives a quite uniform (r-independent) charge distribution for the monolayer. It also minimizes the interactions between a QD inside a circle with its periodic images.

10 Parameters of the iteration

The parameters with control the self consistent iteration cycle are introduced here.

In every iteration loop an effective potential is changes amount of feed back constant. The feedback is set using command

SET feedback 0.2

The maximum number of iterations is set using

SET SCF_MAXITER *n*

where n is the maximum number of iterations. If the changes in the total energy of the system is small enough then the iteration is finished automatically even if the maximum number of iterations are not yet done.

SET SCHRODINGER_EPSILON 0.001

SET SCHRODINGER_MINITER 1

SET SCHRODINGER_MAXITER 10

SET POISSON_MAXITER 40

SET SCHRODINGER_MAXITER 10

SET POISSON_MAXITER 40

11 The local density of states

The density of states is possible to calculate and plot in the file. Using command

DOS *g*

where g is the boarding of the single states gamma peak. The local density of states is possible to plot using command

LDOS *rz g*

where r and z are the place coordinates.

12 The finite temperature

The finite temperature is possible to set on and off using command

SET FINITE_TEMPERATURE .TRUE.

If the finite temperature is on then it is possible to set temperature using command

SET KBT n

where n is k_BT in Hartree units.

13 Some other commands

- **SET broyden_mixing .false.**
- **SET subspace_rotation .false.**
- **SET HARMONIC_OSCILLATOR .false.**
- **SET mg_minschroedinger 4**
- **atom Hoscillator8 0**

14 Output files

- **Z** the coordinates of z-axis.
- **r** the coordinates of r-axis
- **eigenvalues** the eigen energies of the basis functions and the Fermi energy
- **rho_up** electron density for spin up electrons.
- **rho_down** electron density for spin down electrons.
- **rho_p** positive charge density
- **rho_t** total charge density (= rho_up + rho_down -rho_o)
- **V_ionloc** the sum of the local parts of the pseudo potentials
- **V_stab** potential of the stabilized jellium. Constant there where is the stabilized jellium, zero elsewhere.
- **V_up** Potential of the spin up electrons.
- **V_down** Potential of the spin down electrons.
- **V_xc_down** exchange and correlation potential for spin down electrons.
- **V_xc_up** exchange and correlation potential for spin down electrons.
- **V_Gaussian** The potential caused by positive charge.

- **V_initial** First guess for the potential.
- **Total_energy**The total energy and some more information in every iteration cycle.
- **eig_in** The eigenvalues of the single states wave functions.
- **NOS** Number of States
- **POT**
- **OUTPUT** out put file
- **dens_ini** electron density at the begin of the calculation.
- **Efermi** the Fermi energy in Hartree units
- **Weights_up** occupations of the spin up states.
- **Weights_down** occupations of the spin down states.

15 Examples

Here is shown some simple examples of working calculations.

15.1 Three carbon atoms

In the first example there are three carbon pseudo atoms. The boundary conditions are Dirichlet's both r and z directions. During the first RUN, self consistent iterations are done once. After that the multigrid mode is changed to be V_CYCLE and 100 iterations is done.

```
SET MODE START
SET SPIN_COMPENSATED .TRUE.
SET COMPLEX_NUMBERS .FALSE.
SET ENV_ASA .FALSE.
SET ENV_ASA_PERIODIC .FALSE.
SET BOUNDARY_CONDITIONS ZERO
SET WF_FILENAME PSEUDO
SET LOAD_WF .TRUE.
SET SAVE_WF .TRUE.
SET GIVEN_OCCUPATION .FALSE.
SET SCHRODINGER_SOLVER MULTIGRID
SET FINITE_TEMPERATURE .TRUE.
SET KBT 0.0012663
SET BROYDEN_MIXING .FALSE.
SET FEEDBACK 0.05
SET SUBSPACE_ROTATION .FALSE.
SET HARMONIC_OSCILLATOR .FALSE.
SET MAX_M 2
SET M_SUBBANDS 7 4 2
SET K_POINTS 0
SET GRID 63 95
SET CELL 6 12
SET SCHRODINGER_EPSILON 0.001
SET SCHRODINGER_MINITER 1
SET SCHRODINGER_MAXITER 10
SET POISSON_MAXITER 40
SET SCF_MAXITER 1
SET MG_MINSCHROEDINGER 4
ATOM C_tto_rc=1.cpi -4.0
ATOM C_tto_rc=1.cpi 0.0
ATOM C_tto_rc=1.cpi 4.0
SET MULTIGRID_MODE FULL_MULTIGRID
RUN
SET SCHRODINGER_MAXITER 10
SET POISSON_MAXITER 40
SET SCF_MAXITER 100
SET MULTIGRID_MODE V_CYCLE
RUN
QUIT
```

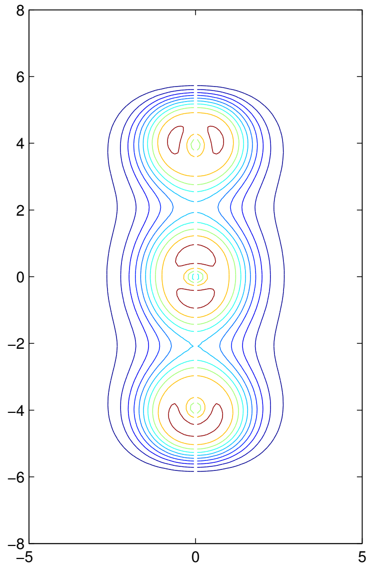

Figure 1: *The electron density from example 1. The plot has an logarithmic scale.*

15.2 Continuation of the calculation

This example continues the previous calculation. It runs 10 more iterations and finally plots some wave functions.

```
SET MODE CONTINUE
SET SPIN_COMPENSATED .TRUE.
SET COMPLEX_NUMBERS .FALSE.
SET ENV_ASA .FALSE.
SET ENV_ASA_PERIODIC .FALSE.
SET BOUNDARY_CONDITIONS ZERO
SET WF_FILENAME PSEUDO
SET LOAD_WF .TRUE.
SET SAVE_WF .TRUE.
SET GIVEN_OCCUPATION .FALSE.
SET SCHRODINGER_SOLVER MULTIGRID
SET FINITE_TEMPERATURE .TRUE.
SET KBT 0.0012663
SET BROYDEN_MIXING .FALSE.
SET FEEDBACK 0.05
SET SUBSPACE_ROTATION .FALSE.
SET HARMONIC_OSCILLATOR .FALSE.
SET MAX_M 2
SET M_SUBBANDS 7 4 2
SET K_POINTS 0
SET GRID 63 95
SET CELL 6 12
SET SCHRODINGER_EPSILON 0.001
SET SCHRODINGER_MINITER 1
SET SCHRODINGER_MAXITER 10
SET POISSON_MAXITER 40
SET SCF_MAXITER 1
SET MG_MINSCHROEDINGER 4
ATOM C_tto_rc=1.cpi -4.0
ATOM C_tto_rc=1.cpi 0.0
ATOM C_tto_rc=1.cpi 4.0
% RUN
SET SCHRODINGER_MAXITER 10
SET POISSON_MAXITER 40
SET SCF_MAXITER 10
SET MULTIGRID_MODE V_CYCLE
RUN
PLOT 0 1
QUIT
```

15.3 Three carbon atoms and jellium leads

In this example there is three carbon atoms and jellium leads. The boundary conditions are periodic both in r and z directions.

```
SET MODE START
SET SPIN_COMPENSATED .TRUE.
SET COMPLEX_NUMBERS .TRUE.
SET BOUNDARY_CONDITIONS PERIODIC
SET ENV_ASA .TRUE.
SET ENV_ASA_PERIODIC .TRUE.
SET WF_FILENAME PSEUDO
SET LOAD_WF .TRUE.
SET SAVE_WF .TRUE.
SET GIVEN_OCCUPATION .FALSE.
SET SCHRODINGER_SOLVER MULTIGRID
SET FINITE_TEMPERATURE .TRUE.
SET KBT 0.0012663
SET BROYDEN_MIXING .FALSE.
SET FEEDBACK 0.05
SET SUBSPACE_ROTATION .FALSE.
SET HARMONIC_OSCILLATOR .FALSE.
SET MAX_M 6
SET M_SUBBANDS 10 8 6 6 4 3 2
SET K_POINTS 2
SET GRID 63 96
SET CELL 6 19.8518
SET SCHRODINGER_EPSILON 0.001
SET SCHRODINGER_MINITER 1
SET SCHRODINGER_MAXITER 10
SET POISSON_MAXITER 40
SET SCF_MAXITER 1
SET MG_MINSCHROEDINGER 4
SET MULTIGRID_MODE FULL_MULTIGRID
SLAB 2.0 -9.9259 -4
SLAB 2.0  4 9.9259
ATOM C_tto_rc=1.cpi -4.0
ATOM C_tto_rc=1.cpi 0.0
ATOM C_tto_rc=1.cpi 4.0
RUN
SET SCHRODINGER_MAXITER 10
SET POISSON_MAXITER 40
SET SCF_MAXITER 100
SET MULTIGRID_MODE V_CYCLE
RUN
QUIT
```

References

[1] R. O. Jones and O. Gunnarsson, Rev. Mod. Phys. **61**, 689 (1989).

[2] J. P. Perdew, H. Q. Tran, and E. D. Smith, Phys. Rev. B **42**, 11627 (1990); H.B. Shore and J.H. Rose, Phys. Rev. Lett. **66**, 2519 (1991).

[3] N. D. Lang, in *Theory of the Inhomogeneous Electron Gas*, edited by S. Lundqvist and N. H. March (Plenum, New York, 1983) p. 309.

[4] M. Koskinen et al., Z. Phys. D **35**, 285 (1995).

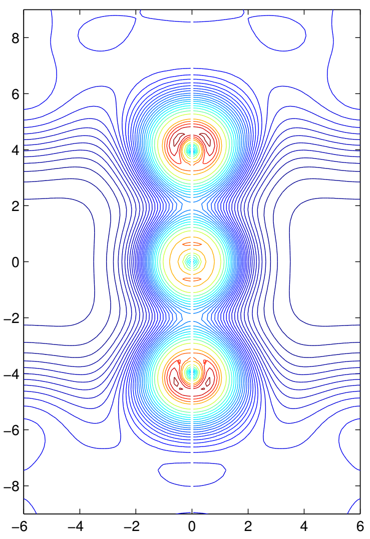

Figure 2: *The electron density from example 1. The plot has an logarithmic scale.*

[5] C. Kittel, *Introduction to Solid State Physics*, 7. edition, pp. 248-252.