The fastest wave

User's manual

Version 2017.03.03

Code name: BlueSKY

NAKANISHI, Hiroshi

National Institute of Technology, Akashi College, Japan

Graduate School of Engineering, Osaka University

You may not give, sell, rent, lease, export or disclose the Naniwa software or any copy of them and this document without permission. Participants of CMD® workshop can use them only for academic research.

In your future publications of academic work performed using NANIWA, the use of the software shall be properly acknowledged, e.g. in the form: "The calculations have been performed using the NANIWA codes developed at Osaka University & National Institute of Technology, Akashi College, Japan [1,2]."

[1] Hiroshi Nakanishi, 2016. Ch.6 Quantum adsorption states of small mass atoms on solid surfaces. In: H. Kasai, M. C. S. Escaño, ed. Physics of Surface, Interface and Cluster Catalysis. Bristol, UK, IOP Publishing.
[2] Hiroshi Nakanishi, Quantum States of the Hydrogen Isotope in Solid Materials and on Their Surfaces", J. Comput. Chem. Jpn., Vol. 15, No. 5, pp. 124–135 (2016).

Contents

1. Introduction

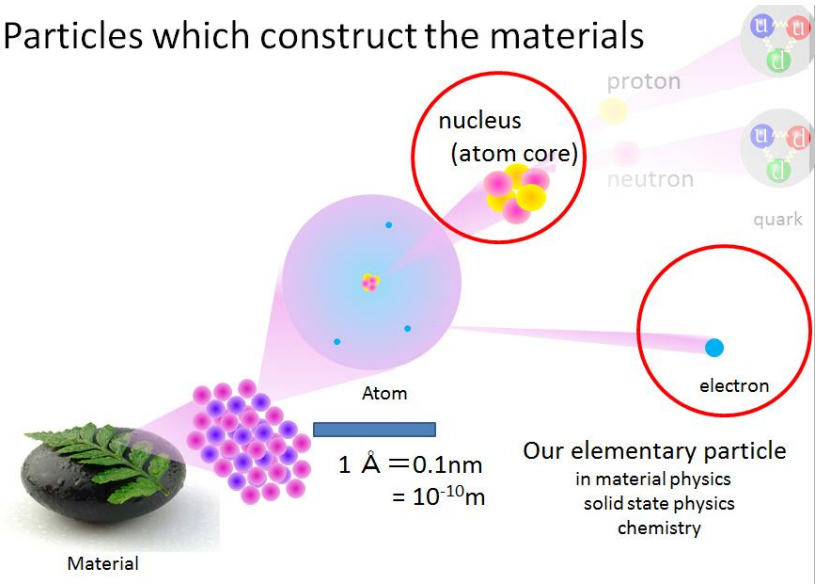
The first principles calculation based on the density functional theory is one of the most successful and powerful methods in the materials physics. And also the first principles molecular dynamics methods have been used for various dynamical behaviors which are from vibrational motions to chemical or biochemical reactions. In the case of small mass atoms like hydrogen atoms, we should treat their nuclei in a quantum-mechanical manner, in addition to the electrons. The quantum behaviors of the nuclei have crucial roles to play in the materials related to today's energy technology: fuel cell and lithium-ion rechargeable battery, and also various catalytic reactions to produce hydrogen fuel, not only from their academic interests.

We have been developing the quantum simulation code for the small mass atoms motion in materials, "Naniwa" [1-15].

"NANIWA" series codes are computational codes for performing first principles quantum mechanical calculations, which were developed at Kasai laboratory (-2015) in Osaka University and have been developed at Nakanishi laboratory in NITAC (2016-). This document describes about the *Naniwa-K* [3,4], which is a nucleus version of the first principles quantum state calculations [5-15]. You can solve the eigenvalue problem and obtain the eigenstates and their eigenenergies for atom (nuclear) motion in the materials by this code.

1.1 Naniwa formulation

We consider that kinds of particles which construct the material are electron and nucleus. If we know how these particles behave, we can know the material properties.



Total Hamiltonian of the electrons and nuclei in material is given by

$$H = \sum_{I=1}^{N_{\mathrm{nc.}}} \left(-\frac{\hbar^2}{2M_I} \right) \nabla_I^2 + \sum_{i=1}^{n_e} \left(-\frac{\hbar^2}{2m_{\mathrm{e}}} \right) \nabla_i^2 + V(\vec{r}, \vec{R}) \tag{1}$$

where M_I and m_{e} are masses of electron and nucleus I, and $\hbar$ is the Dirac constant. Numbers of nucleus and electron are $N_{\mathrm{nc.}}$ and $n_{\mathrm{e.}}$. Potential energy is given by the coulomb interaction between all particles;

$$V(\vec{r},\vec{R}) = \sum_{i=1}^{n_{\mathrm{e}}} \sum_{I=1}^{N_{\mathrm{nc.}}} \frac{-Z_I e^2}{\left|\vec{r}_i - \vec{R}_I\right|} + \sum_{i=1}^{n_{\mathrm{e}}} \sum_{j=1}^{i-1} \frac{e^2}{\left|\vec{r}_j - \vec{r}_i\right|} + \sum_{I=1}^{N_{\mathrm{nc.}}} \sum_{J=1}^{I-1} \frac{Z_J Z_I e^2}{\left|\vec{R}_J - \vec{R}_I\right|} \tag{2}$$

where Z_I is atomic number of nucleus I, and e is elementary charge.

The positions of the electrons and nuclei are indicated by

$$\vec{r} = \left(\vec{r}_1, \vec{r}_2, \vec{r}_3, \cdots, \vec{r}_i, \cdots \vec{r}_{n_{\mathrm{e}}}\right), \tag{3.1}$$

$$\vec{R} = \left(\vec{R}_1, \vec{R}_2, \vec{R}_3, \cdots, \vec{R}_I, \cdots, \vec{R}_{N_{\mathrm{nc.}}}\right). \tag{3.2}$$

Total Schrödinger equation is

$$H\,\Psi(\vec{r},\vec{R}) = E\,\Psi(\vec{r},\vec{R}), \tag{4}$$

where E and Ψ are the total energy and wave function describing the state of this material system.

Within the adiabatic approximation, we can divide the wave function Ψ into two parts;

$$\Psi_{n,\omega}(\vec{r},\vec{R}) = \psi_{n;\vec{R}}(\vec{r}) \cdot \phi_{\omega;n}(\vec{R}), \tag{5}$$

Where $\psi_{n;\vec{R}}(\vec{r})$ is the n-th electron wave function in the case of the fixed nucleus position $\vec{R}$ and $\phi_{\omega;n}(\vec{R})$ is the ω-th nucleus motion wave function in the case of the electron state n.

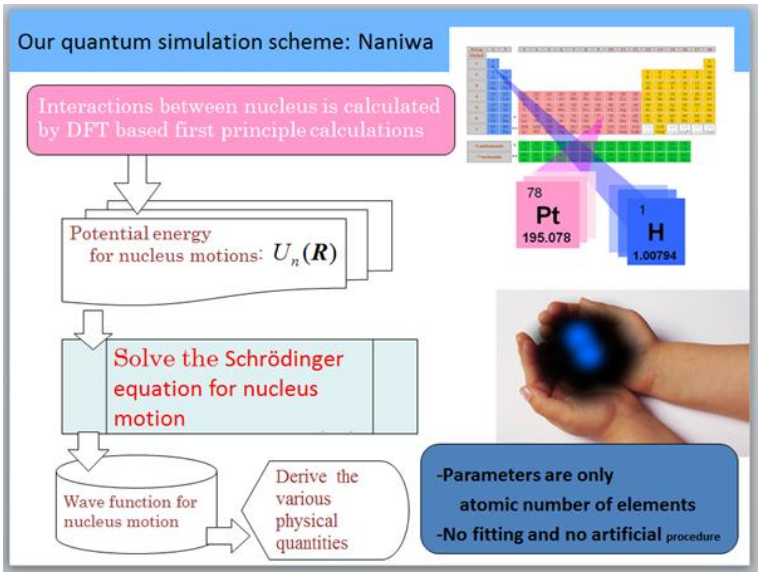
Then, we can divide the total Schrödinger equation into two equations. One is for electrons,

$$\left[\sum_{i=1}^{n_e} \left(-\frac{\hbar^2}{2m_e} \right) \nabla_i^2 + V(\vec{r},\vec{R}) \right] \psi_{n;\vec{R}}(\vec{r}) = U_n(\vec{R})\, \psi_{n;\vec{R}}(\vec{r}) \tag{6}$$

Another is for nuclei

$$\left[\sum_{I=1}^{N_{nc}} \left(-\frac{\hbar^2}{2M_I} \right) \nabla_I^2 + U_n(\vec{R}) \right] \phi_{\omega;n}(\vec{R}) = E_{\omega;n} \phi_{\omega;n}(\vec{R}). \tag{7}$$

From Schrödinger equation for electron system with fixed *R*, we can obtain the eigenenergies and eigenstates with the aid of the conventional first principles (*electron states*) calculation. The eigenenergies, $U_n(\vec{R})$ of electron state as a function of *R* can be considered as the adiabatic potential energy surface for nucleus motion. Eigenenergy of nuclei corresponds to the total energy, *E,* appeared in the Schrödinger equation for total system. Naniwa series codes solve this equation.



Naniwa-SKY code solves the Schrödinger equation for a single atom motion in the material.;

$$\left[-\frac{\hbar^2}{2M}\left(\frac{\partial^2}{\partial X^2}+\frac{\partial^2}{\partial Y^2}+\frac{\partial^2}{\partial Z^2}\right)+U_0(X,Y,Z)\right]\phi_{\omega;0}(X,Y,Z)=E_{\omega,0}\phi_{\omega;0}(X,Y,Z). \tag{8}$$

The wave function for atom motion in the material has rather position localized character. Then we described it by linear combination of the 3D-Gauss functions located at grid points,

$$\phi_\omega(X,Y,Z)=\sum_{\ell=1}^{N_G}C_{\omega,\ell}G_\ell(X,Y,Z), \tag{9}$$

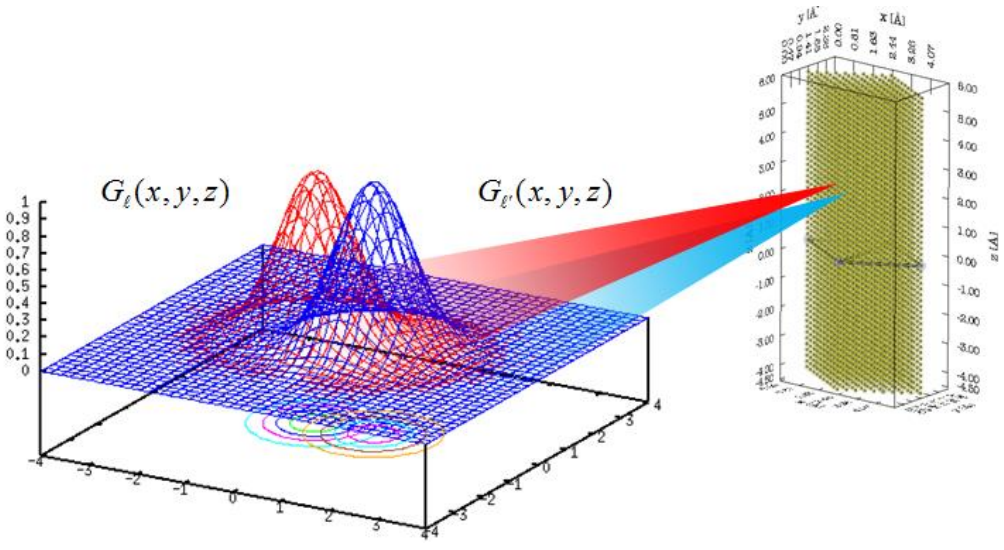
where $G_\ell(X,Y,Z)$ is the ℓ-th 3D-Gauss function located at the ℓ-th grid point $\vec{R}_\ell(X_\ell,Y_\ell,Z_\ell)$,

$$G_\ell(X,Y,Z)=\left(\frac{8\alpha_X\alpha_Y\alpha_Z}{\pi^3}\right)^{1/4}\exp\left\{-\alpha_X(X-X_\ell)^2-\alpha_Y(Y-Y_\ell)^2-\alpha_Z(Z-Z_\ell)^2\right\}, \tag{10}$$

Factors $\alpha_X,\alpha_Y,\alpha_Z$ indicate the decays of Gauss function along the X, Y, Z directions

$$G_\ell(X,Y,Z;\vec{k})=\left(\frac{8\alpha_X\alpha_Y\alpha_Z}{\pi^3}\right)^{1/4}\exp\left\{-\alpha_X(X-X_\ell)^2+ik_XX\quad-\alpha_Y(Y-Y_\ell)^2+ik_YY-\alpha_Z(Z-Z_\ell)^2+ik_ZZ\right\}$$

The nearest neighbor 3D-Gauss functions have to be overlapped.



Naniwa code use the variation method with the trial function (9) to solve the Schrödinger equation (8). And Naniwa code solves the following equation instead of eq (8).

$$\sum_{\ell'}\left[H_{\ell\ell'}-E_{\omega,0}S_{\ell\ell'}\right]C_{\omega,\ell}=0 \tag{11.0}$$

$$H_{\ell\ell'}=K_{\ell\ell'}+U_{\ell\ell'} \tag{11.1}$$

$$K_{\ell\ell'}=\iint\int_{-\infty}^{+\infty}G_{\ell}(X,Y,Z)\left[-\frac{\hbar^2}{2M}\left(\frac{\partial^2}{\partial X^2}+\frac{\partial^2}{\partial Y^2}+\frac{\partial^2}{\partial Z^2}\right)\right]G_{\ell'}(X,Y,Z)dXdYdZ, \tag{11.2}$$

$$U_{\ell\ell'}=\iint\int_{-\infty}^{+\infty}G_{\ell}(X,Y,Z)U_0(X,Y,Z)\,G_{\ell'}(X,Y,Z)dXdYdZ \tag{11.3}$$

$$S_{\ell\ell'}=\iint\int_{-\infty}^{+\infty}G_{\ell}(X,Y,Z)\,G_{\ell'}(X,Y,Z)dXdYdZ \tag{11.4}$$

Here, we can integrate K and S, analytically. On Naniwa-K code, the special routines are installed to integrate U in an efficient way [Japanese Paten No.5902495]. From eq.11.0, secular equation is written by

$$\begin{vmatrix} H_{11}-E\,S_{11} & H_{12}-E\,S_{12} & \cdots & H_{1N_G}-E\,S_{1N_G} \\ H_{21}-E\,S_{21} & H_{22}-E\,S_{22} & \cdots & H_{2N_G}-E\,S_{2N_G} \\ \vdots & \vdots & \ddots & \vdots \\ H_{N_G1}-E\,S_{N_G1} & H_{N_G2}-E\,S_{N_G2} & \cdots & H_{N_GN_G}-E\,S_{N_GN_G} \end{vmatrix}=0 \quad . \tag{12}$$

We solve the secure eq.(12) and obtain eigenenergy $E_{\omega,0}$, $\omega=0,1,2\cdots$. Then, we can solve eq.(11.0) for all obtained $E_{\omega,0}$, and get the corresponding eigenvectors $(C_{\omega,1},C_{\omega,2},C_{\omega,3},\cdots,C_{\omega,N_G})$.

You should check the convergence of these numerical results by the some trial runs for various Gaussian decay factors $\alpha_X,\alpha_Y,\alpha_Z$ and number of grid point, N_G.

The expectation value of an observable, O, at ωth state is given by

$$\begin{aligned}\langle O\rangle_{\omega}&=\iiint\phi_{\omega}^{*}(X,Y,Z)\hat{O}\phi_{\omega}(X,Y,Z)\,dXdYdZ\\&=\sum_{q=1}^{N_G}\sum_{q'=1}^{N_G}C_{\omega,q}^{*}C_{\omega,q'}\iiint G_q(X,Y,Z)\hat{O}G_{q'}(X,Y,Z)\,dXdYdZ\end{aligned} \quad . \tag{13}$$

It is easy to calculate $\langle O\rangle_{\omega}$ from obtained eigenvectors$(C_{\omega,1},C_{\omega,2},C_{\omega,3},\cdots,C_{\omega,N_G})$.

2. System Requirements

Hardware

Recommended

Computer Processor: 3.0GHz Intel Core i7 or better

Computer Memory: 8 GB/core or more

Operating System: 64-bit Linux distribution

Software

(1) Naniwa package file : NaniwaSKYXXXXXXXX.tar.gz

XXXXXXXX is version number. (ex. NaniwaSKY20160817.tar.gz)

(2) ab initio electronic state calculation package

For install

(1) Compiler: Intel® Fortran compiler [16]

(2) Math library: Intel ® Math Kernel Library (MKL) [17]

For use

(1) Text editor: vi , mule … (as you like)

(2) C shell: csh or tcsh

(When you use other Unix shell, you must rewrite the script files by yourself.)

(3) Visualization tool: gnuplot and OpenDX, XCrySDen, or VESTA.

"gnuplot" is a command-line program that can make 2- and/or 3-dimensional plots of functions and data. [18]

"**OpenDX**" is IBM Visualization Data Explorer program. [19]

"**XCrySDen**" is a crystalline and molecular structure visualization program. [20]

"VESTA" is a Visualization program for Electronic and STructural Analysis.

3. How to install

(1) Copy the package file “NaniwaSkyXXXXXXXX.tar.gz” to your home directory.

XXXXXXXX is version number. (ex. NaniwaSky20160817.tar.gz)

(2) Decompress the package file.

Type following command lines:

gzip -d NaniwaSkyXXXXXXXX.tar.gz [Enter]

tar xvf NaniwaSkyXXXXXXXX.tar [Enter]

You can get following directory on your home directory.

Installed directory structure

```
~-+  : your home directory
   |-- naniwa -+
               |-- doc:  documents
               |-- SRC :          source codes
               |-- bin: :you will install the execution programs into this directory
               |-- work:          working directory
               |-- qs :  script for Grid Engine (Job scheduler)
               |-- etc -+
                        |-- OpenDX:       files for Visualization software OpenDX®
                        |-- charge:               files for BADER CHARGE ANALYSIS
                        |-- potential:    files for making potential data
                        |-- example:      input data examples
                        |-- results:              some simulation results
```

(3) Move to SRC directory

Type :

cd ./naniwa/SRC [Enter]

(4)Compile the source codes

Type :

make all [Enter]

The compilation must finish without errors, although warnings may be possible.

If you get error message, you must modify following values in “Makefile” file under SRC directory.

(A) Error message

```
make: ifort: Command not found
make: *** [naniwa] Error 127
```

Please ask to your system administrator to install Intel Fortran.

If you use older Intel Fortran, please change the compiler name in " Makefile"

From

FC = ifort

To

FC = ifc

(B) Error message:

```
/tmp/ifortzvGoSf.o: In function `MAIN__':
NaniwaSYMk.f:(.text+0x40fb): undefined reference to `dfti_create_descriptor_highd'
NaniwaSYMk.f:(.text+0x410f): undefined reference to `dfti_set_value_intval'
NaniwaSYMk.f:(.text+0x4123): undefined reference to `dfti_set_value_intval'
NaniwaSYMk.f:(.text+0x4137): undefined reference to `dfti_set_value_intvec'
NaniwaSYMk.f:(.text+0x414b): undefined reference to `dfti_set_value_intvec'
NaniwaSYMk.f:(.text+0x4155): undefined reference to `dfti_commit_descriptor_external'
NaniwaSYMk.f:(.text+0x416d): undefined reference to `dfti_compute_forward_zz'
NaniwaSYMk.f:(.text+0x41a0): undefined reference to `dfti_free_descriptor_external'
NaniwaSYMk.f:(.text+0x6fb2): undefined reference to `zhegv_'
make: *** [naniwa] Error 1
```

If you see the above-mentioned errors, you must change the path of Intel MKL installed directory in Makefile

From

LIB = /opt/intel/mkl /lib/intel64

INC= -I/opt/intel/mkl/include

To

LIB = /opt/intel/mkl /lib/em64t

INC= -I/opt/intel/mkl/include

or

LIB = /opt/intel/mkl/XXXXXX/lib/em64t

INC = -I/opt/intel/mkl/XXXXXX/include

here, XXXXXXX is version of the Intel MKL installed on your computer.

(C) Error message:

```
fastDFT3D_MKL.f(5): error #7002: Error in opening the compiled module file.  Check INCLUDE paths.  [MKL_DFTI]
      Use MKL_DFTI
----------^
fastDFT3D_MKL.f(10): error #6457: This derived type name has not been declared.  [DFTI_DESCRIPTOR]
      type(DFTI_DESCRIPTOR), POINTER :: hand
-----------^
compilation aborted for fastDFT3D_MKL.f (code 1)
make: *** [naniwa] Error 1
```

If you see the above-mentioned errors, you should ask to your system administrator to compile mkl_dfti.f90.

Please do followings as root (super user).

cd /opt/intel/mkl/include

ifort –c mkl_dfti.f90

* You can find some hints to solve your problem in Makefile.

(5) Move the execution file to binary directory

Type:

make all-install [Enter]

Check the execution files under bin directory

Type :

ls ../bin [Enter]

You can see following files:

BandStructure	cellexpander	chkPOTENG	makeBHGCAR	naniwaSS_run	state2dx
BHGCAR2DX	ChargeState	eigen2spec	makePOTENG	poteng2dx	state2xsf

(6) test run.

Move to work directory.

cd [Enter]

cd ./naniwa/work [Enter]

Copy the test data.

cp -r ../etc/example/test ./ [Enter]

cd ./test [Enter]

If necessary, change the "LD_LIBRARY_PATH" value in "run.csh" file by text editor :

setenv LD_LIBRARY_PATH LD_LIBRARY_PATH:'/opt/intel/mkl/lib/em64t'.

or

setenv LD_LIBRARY_PATH LD_LIBRARY_PATH:'/opt/intel/mkl/ XXXXXXX /lib/em64t'.

here, XXXXXXX is version of the Intel MKL installed on your computer.

Execute the program

csh ./run.csh [Enter]

machine name

Job start date & time

Naniwa start date & time

finish naniwa

Naniwa terminate date & time

Check output files

ls [Enter]

```
INSET   kpoint0000   LOG   POTENG   run.csh
```

If successful, you can get the following LOG file:

```
****************************************
 *              Naniwa code              *
 ****************************************
     NaniwaSky Version 2012.11.22
            created by NAKANISHI, Hiroshi
            OSAKA UNIVERSITY

  ----------------------------------------
   INSET title :
 *


   Calculaton mode :                0
   *** Full calculation
  ----------------------------------------
   Atomic Mass =     1.00000000000000          (proton)
                 =   1.672621637000000E-027 (kg)
  ----------------------------------------
  Number of node index (No1,No2,No3)
               9               9               9
  Gaussian decay factors (1/Angstrom Angstrom)
  SigmaX=     5.00000000000000
  SigmaY=     5.00000000000000
  SigmaZ=     5.00000000000000
  ----------------------------------------
  ** No correction by symmetry.
  ----------------------------------------
  - Lattice vector (A)
        3.00000000000000                    0.00000000000000
0.000000000000000E+000
      0.000000000000000E+000                3.0000000
0.000000000000000E+000
      0.000000000000000E+000                0.00000000000000
3.00000000000000
   - Lattice unit volume (A^3)
         27.0000000000000
  ----------------------------------------
   Reciprocal lattice vector (/A)
        2.09439510239320                    0.00000000000000
0.000000000000000E+000
      0.000000000000000E+000                2.0943951
0.000000000000000E+000
      0.000000000000000E+000                0.00000000000000
2.09439510239320
```

```
wave00005.xsf    was created.
WAVE function
    Max value=     1.68540114820597
    Min value=    -1.84336513692804
 probability=    0.999869835826559


wave00006.xsf    was created.
WAVE function
    Max value=     1.93274716286747
    Min value=    -1.03123097291435
 probability=    0.999869796584020


wave00007.xsf    was created.
WAVE function
    Max value=     1.72099225797871
    Min value=    -1.72768364358320
 probability=     1.00175769772228


wave00008.xsf    was created.
WAVE function
    Max value=     1.00317921937198
    Min value=    -1.96774733538390
 probability=     1.00175785191399


wave00009.xsf    was created.
WAVE function
    Max value=    0.825426714232840
    Min value=    -2.44372845893816
 probability=     1.00175785169862


wave00010.xsf    was created.
WAVE function
    Max value=     1.64599264710982
    Min value=    -1.54178274285677
 probability=    0.999839014572012
```

4. Program codes and files

4.1 Programs

(1) naniwaSS_run : main simulation execution file to calculate eigenstates and eigenenergies

Input files: INSET : simulation setting information

POTENG : Potential energy surface for atom motion

KPOINTS (option): for dispersion [Under construction, Coming soon]

Output files: LOG : run time information

EIGEN: Eigenenergy for atom motion

stateXXXXX: information about eigenstate for atom motion

(2) Tools for getting potential energy surface for atom motion

(2.1) auto.csh: automatic launcher for ab initio electronic state calculation VASP

(2.2) collect.csh: to make POTDAT from many OSZICAR files of VASP

(2.3) makePOTENG: to make POTENG from POTDAT and POTDAT.cfg

(2.4) chkPOTENG: checking the max and min values in POTENG file.

(3) Tools for visualization of results

(3.1) poteng2dx: to make input file for WaveViewer3D in order to see the potential energy surface

input file: POTENG, POTDAT.cfg (option)

output file: PES.dat, PES3D.general

(3.2) state2dx: to make input file for WaveViewer3D in order to see the wave function

input files: stateXXXXX, POTDAT.cfg (option)

output files: waveXXXX.dat, waveXXXX.general

usage : state2dx [start state number] [end state number]

example: state2dx 1 10 *[Enter]*

(3.3) WaveViewer3D_XXX (Visual program for OpenDX (http://www.opendx.org/)

WaveViewer3D_XXX.cfg, WaveViewer3D_XXX.net

(3.4) state2xsf: to make input file for XcrySDen in order to see the wave function

input files: stateXXXXX, POTDAT.cfg (option)

output files: waveXXXX.xsf

usage : state2xsf [start state number] [end state number]

example: state2xsf 1 10 *[Enter]*

(4) Band structure builder

(4.1) BandStructure: [Under construction, coming soon]

4.2 files

4.2.1 List of important files

INPUT files:

INSET	:	Calculation setting about basis function and particle mass, etc.
POTENG	:	Super cell geometry and potential energy surface
KPOINTS	:	[Under construction, coming soon]

OUTPUT files:

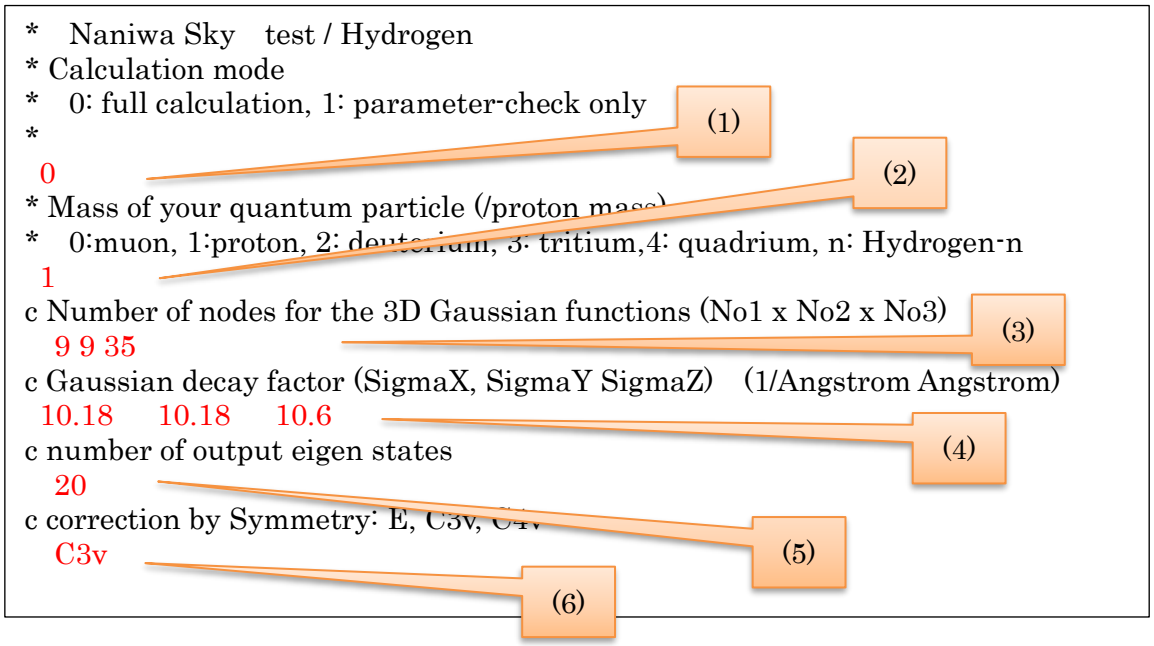
LOG	:	calculation information

under the directory of "kpoint0000"

EIGEN	:	Eigenenergy set
state00000	:	Eigenstate information of the ground state
stateXXXXX	:	Eigenstate information of the XXXXXX th excited state

4.2.2 INSET

Example



Contents

(0) Comment lines

The lines that begin with with a "c" or "*" (asterisk) are comment line. In comment lines, you can write something as you like in these lines, but don't change the number of all comment lines and their positions in INSET.

(1) Calculation mode in the 5th line

0; full calculation mode.

1: parameter check mode. (no calculation)

(2) Mass of the target quantum particle, which unit is that of a proton.

1: proton, 2: deuterium, 3: tritium,4: quadrium, n: Hydrogen-n

if you input 0, program will replace it by the mass value of the (Anti)muon.

(3) Number of nodes for the 3D Gaussian functions (No1 x No2 x No3) in th 8th line

Please input 3 positive integer numbers which are numbers of nodes for 3D Gaussian functions, No1, No2, No3. They correspond to the numbers of partitions of primitive vectors: $L\overrightarrow{a_1}, \overrightarrow{La_2}, L\overrightarrow{a_3}$ (See next section 4.2.3)

(4) Gaussian decay factor (SigmaX, SigmaY SigmaZ) in the 10th lines

Please input 3 positive floating-point type data which are Gaussian decay factors along the directions of x, y and z-axis, σ_x, σ_y, σ_z. Their unit is 1/(Angstrom・Angstrom). (If you put 0, 0, 0, then these values are automatically reset by default values.)

(5) Number of output eigen states in the 12th line

Please input a positive integer number of output eigen states which you request.

(6) Correction by crystal structure symmetry

(6)-1. Correction by rotational symmetry around z-axis

Please input one rotational symmetry symbol.

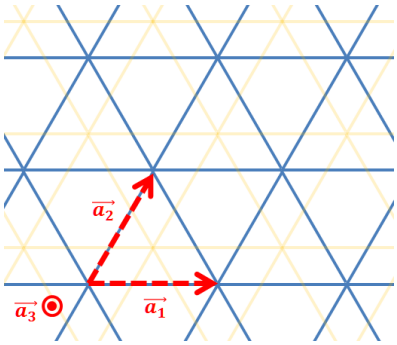
Supported symbols: C1, C3v, C4v (Schönflies notation)
for surfaces perpendicular to Z-axis

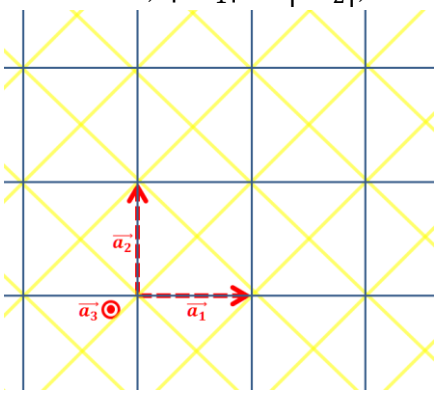
(6)-2. Correction by Crystal structure symmetry

Supported symbols: Oh, Td^2, Oh^7 (Schönflies notation)
graphene (special)

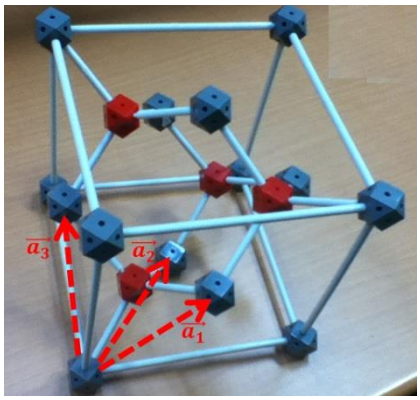
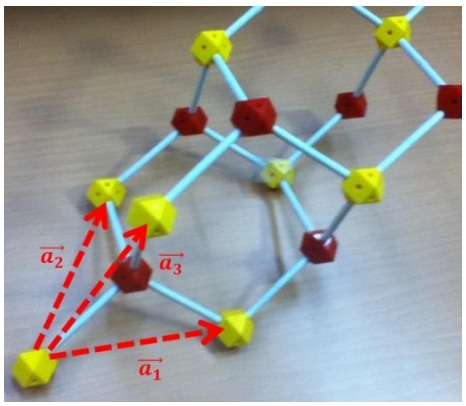
Supported symmetry symbols.

(i)	
Symbol	E
Symmetry	1-fold rotation
Requirement	non

(ii)	
Symbol	C3v
Symmetry	3-fold rotation symmetry with 3 mirror planes containing the z-axis (vertical planes).
Requirement	No1=No2, $\|L\overrightarrow{a_1}\| = \|\overrightarrow{La_2}\|$, $2\overrightarrow{a_1}\cdot\overrightarrow{a_2} = \|\overrightarrow{a_2}\|^2$, $\sigma_x = \sigma_y$ 

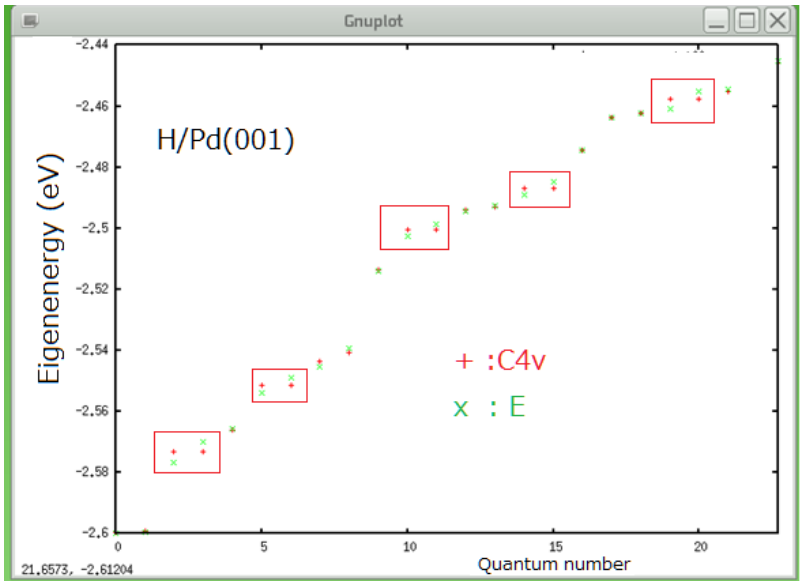
(iii)	
Symbol	C4v
Symmetry	4-fold rotation symmetry with 4 mirror planes containing the z-axis (vertical planes).
Requirement	No1=No2, $\|L\overrightarrow{a_1}\| = \|\overrightarrow{La_2}\|$, $\sqrt{2}\,\overrightarrow{a_1}\cdot\overrightarrow{a_2} = \|\overrightarrow{a_2}\|^2$, $\sigma_x = \sigma_y$ 

(iv)	
Symbol	Oh
Symmetry	The rotation axes of an octahedron or cube (three 4-fold axes, four 3-fold axes, and 6 diagonal 2-fold axes) with a mirror plane perpendicular to the z-axis (horizontal plane).
Requirement	No1=No2=No3, $\lvert L\overrightarrow{a_1}\rvert = \lvert\overrightarrow{La_2}\rvert = \lvert\overrightarrow{La_3}\rvert$, $\sqrt{2}\,\overrightarrow{a_1}\cdot\overrightarrow{a_2} = \sqrt{2}\,\overrightarrow{a_1}\cdot\overrightarrow{a_3} = \sqrt{2}\,\overrightarrow{a_2}\cdot\overrightarrow{a_3} = \lvert\overrightarrow{a_2}\rvert^2$, $\sigma_x = \sigma_y = \sigma_z$

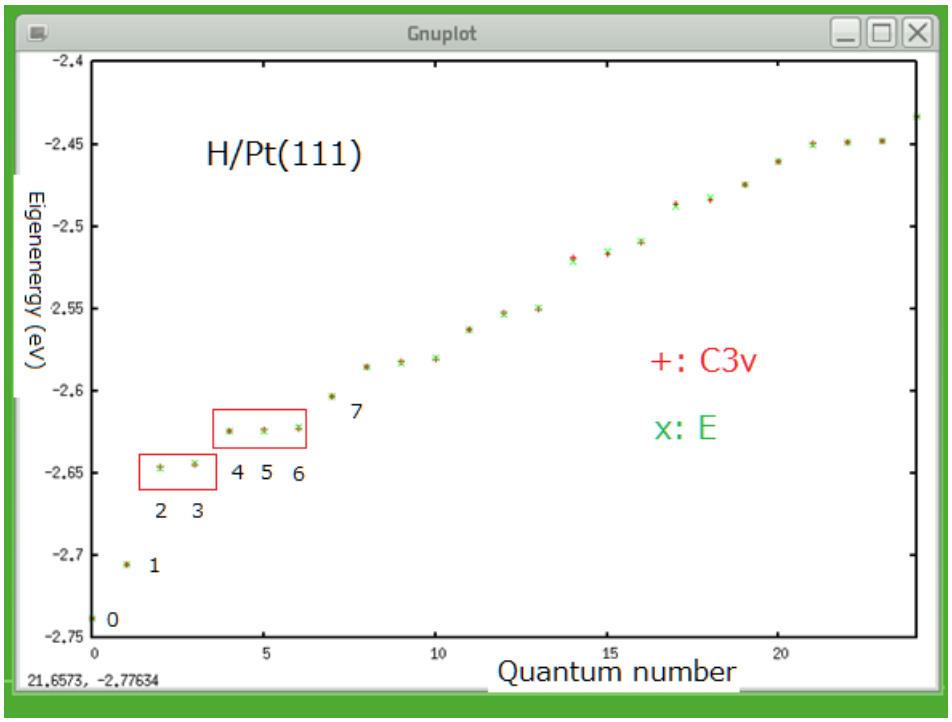
(v)	
Symbol	Td^2, F -4 3m, 216
Symmetry	Zinc Blend crystal structure.
Requirement	No1=No2=No3, $\lvert L\overrightarrow{a_1}\rvert = \lvert\overrightarrow{La_2}\rvert = \lvert\overrightarrow{La_3}\rvert$, $2\,\overrightarrow{a_1}\cdot\overrightarrow{a_2} = 2\,\overrightarrow{a_1}\cdot\overrightarrow{a_3} = 2\,\overrightarrow{a_2}\cdot\overrightarrow{a_3} = \lvert\overrightarrow{a_1}\rvert^2$, $\sigma_x = \sigma_y = \sigma_z$  

(vi)	
Symbol	Oh^7, F d -3 m, 227
Symmetry	Diamond crystal structure.
Requirement	No1=No2=No3, $\lvert L\overrightarrow{a_1}\rvert = \lvert\overrightarrow{La_2}\rvert = \lvert\overrightarrow{La_3}\rvert$, $2\,\overrightarrow{a_1}\cdot\overrightarrow{a_2} = 2\,\overrightarrow{a_1}\cdot\overrightarrow{a_3} = 2\,\overrightarrow{a_2}\cdot\overrightarrow{a_3} = \lvert\overrightarrow{a_1}\rvert^2$, $\sigma_x = \sigma_y = \sigma_z$ No1 mod 4 = 0

When you use this function, you can get accurate degenerate eigen values.
In the case of hydrogen atom in Pd(001), which has C4v symmetry, you can get following eigenvalues:



And in the case of hydrogen atom in Pt(111), which has C3v symmetry, you can get following eigenvalues:

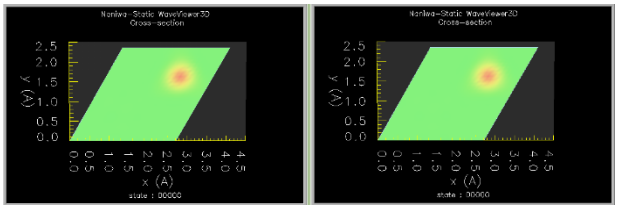


Green: no correction

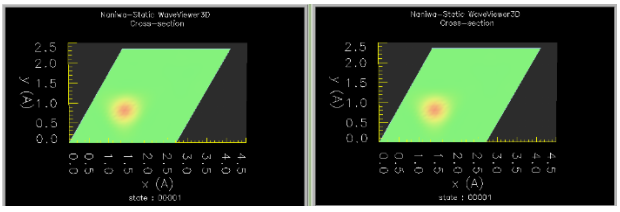
and their wave functions:

E (no use) C3v

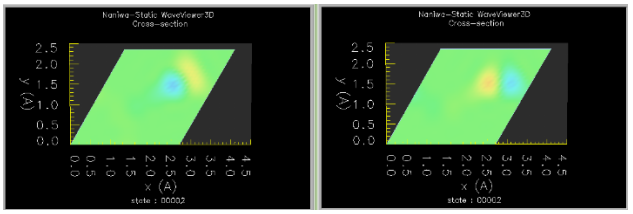
the ground state



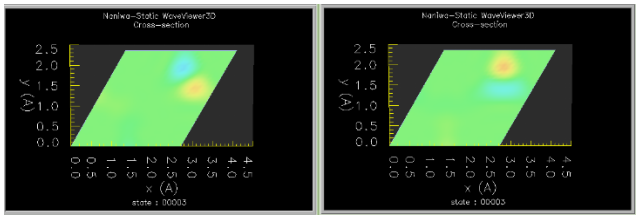
the 1st excited state



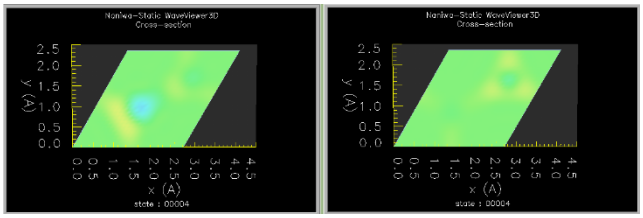
the 2nd excited state



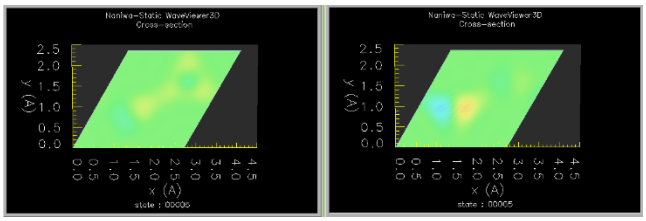
the 3rd excited state



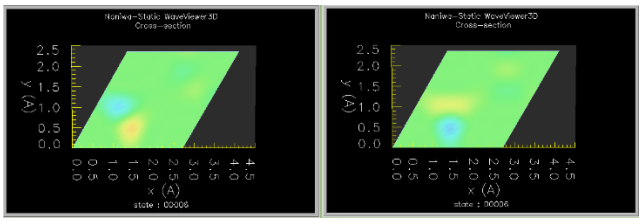
the 4th excited state

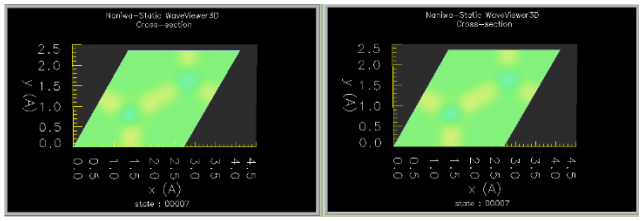


the 5th excited state



the 6th excited state



the 7th excited state 

4.2.3 POTENG

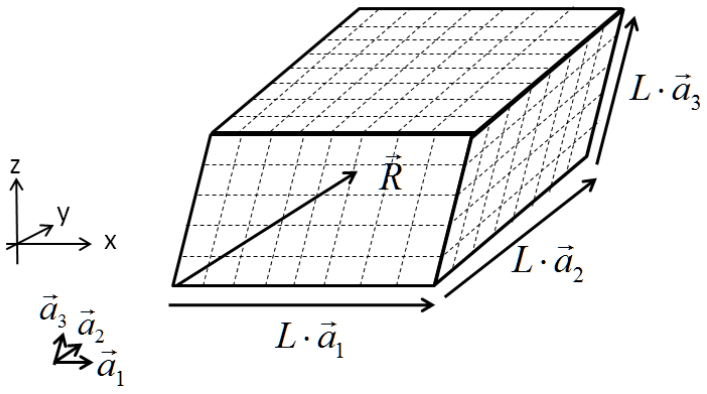
The first line is a comment line. The second line provides a universal scaling factor L (Angstrome), which is used to scale the primitive vectors in the simulation system. On the following three lines, the three primitive vectors, $L\overrightarrow{a_1}$, $L\overrightarrow{a_2}$, $L\overrightarrow{a_3}$,defining the supercell of the simulation material system are given . Each line has the x, y, z components of Cartesian coordinate system, whose unit is L. The sixth line supplies the numbers, N_1, N_2, N_3, of the potential energy calculation point along the primitive vectors, $\overrightarrow{a_1}$, $\overrightarrow{a_2}$, $\overrightarrow{a_3}$. The lines from seventh provide the potential energy values (eV) and its particle positions. Each line has $\mathrm{n}_1, \mathrm{n}_2, \mathrm{n}_3, U(R_{\mathrm{n}_1,\mathrm{n}_2,\mathrm{n}_3})$.

$$\text{Here, } R_{\mathrm{n}_1,\mathrm{n}_2,\mathrm{n}_3} = L\left(\frac{\mathrm{n}_1}{N_1}\overrightarrow{a_1} + \frac{\mathrm{n}_2}{N_2}\overrightarrow{a_2} + \frac{\mathrm{n}_3}{N_3}\overrightarrow{a_3}\right),$$

where

$$n_1 = 0,1,2,\cdots,N_1 - 1$$
$$n_2 = 0,1,2,\cdots,N_2 - 1$$
$$n_3 = 0,1,2,\cdots,N_3 - 1$$

.



```
* POTENG: POTENTIAL ENERGY for H on Pd(111)
    2.820295
    1.000000          0.0000000E+00   0.0000000E+00
   0.5000000          0.8660254         0.0000000E+00
   0.0000000E+00   0.0000000E+00     3.81165754909725
             12              12              44
              1               1               1 -8.7710377E-04
              2               1               1 -6.4631744E-04
              3               1               1 -3.8508927E-03
              4               1               1 -1.8527496E-03
              5               1               1 -2.3826952E-03
              6               1               1 -4.5110146E-03
              7               1               1 -6.5468652E-03
              ~               ~                ~

              ~               ~                ~
             11               2               1 -1.1861493E-03
             12               2               1 -6.4631744E-04
              1               3               1 -3.8508927E-03
              2               3               1 -1.2586723E-03
              3               3               1 -1.6357267E-03
              4               3               1 -2.8395073E-03
              5               3               1 -5.4807067E-03
              6               3               1 -8.3559640E-03
              2              12              44   -1.732900
              3              12              44 -0.2266000
              4              12              44     1.000000
              5              12              44     1.000000
              6              12              44     1.000000
              7              12              44     1.000000
              8              12              44     1.000000
              9              12              44     1.000000
             10              12              44     1.000000
             11              12              44     1.000000
             12              12              44 -0.2266000
 *    END of POTENG
```

4.2.4 LOG

It gives you the information about your simulation situation.

When the simulation is executing, you can check it by following command.

tail -f LOG [Enter]

Following is the example of LOG file.

```
****************************************
 *              Naniwa code                *
 ****************************************
 mode: NaniwaSSk Version 2012.02.10
             created by NAKANISHI, Hiroshi
             OSAKA UNIVERSITY

 ----------------------------------------
  Calculaton mode :              0
  *** Full calculation
 ----------------------------------------
```

```
 Atomic Mass =      1.00000000000000          (proton)
                  =   1.672621637000000E-027 (kg)
 ----------------------------------------
 Number of node index (No1,No2,No3)
               9               9              35
 Gaussian decay factors (1/Angstrom Angstrom)
 SigmaX=     5.09173682192627
 SigmaY=     5.09173682192627
 SigmaZ=     8.55607353478694
 ----------------------------------------
 ----------------------------------------
 - Lattice vector (A)
       3.98849944000000          0.000000000000000E+000    0.000000000000000E+000
     0.000000000000000E+000      3.98849944000000          0.000000000000000E+000
     0.000000000000000E+000    0.000000000000000E+000      11.9654983200000
 - Lattice unit volume (A^3)
        190.348676260400
 ----------------------------------------
 Reciprocal lattice vector (/A)
       1.57532565230225          0.000000000000000E+000    0.000000000000000E+000
     0.000000000000000E+000      1.57532565230225          0.000000000000000E+000
     0.000000000000000E+000    0.000000000000000E+000    0.525108550767416
 Reciprocal lattice unit volume (/A^3)
        1.30313611328161
 ----------------------------------------
----------------------------------------
 MSG: reading POTENG
 MSG: finish reading POTENG
 MSG: start FFT of POTENG data
 MSG: finish FFT
  I cannot find KPOINTS file.
  I will set    single k-point calculation.
 -----------------------------------------------------
 K-sampling point information.
 Number of K-points:                  1
 ***
  # fractional of reciprocal lattice vector
                          (unit : b1/2, b2/2, b3/2)
               1 :    0.0000000E+00    0.0000000E+00    0.0000000E+00
 ***
  # kx,ky,kz vector (/A)
               1 :    0.0000000E+00    0.0000000E+00    0.0000000E+00
 -----------------------------------------------------
 MSG: start k-loop
 +++++++++++++++++++++++++++++++++++++++++++++++++++
 MSG: k-loop:                  1

 MSG:    POTMTX file is created
 ----------------------------------------
 MSG: start diagonalization
 ----------------------------------------
 MSG: end diagonalization
 MSG: info from ZHEGV =                    0
 ----------------------------------------
 ----------------------------------------
 Eigen energy value (eV)
               0    -2.738340
               1    -2.705202
               2    -2.647121
               3    -2.643604
               4    -2.624361
               5    -2.624101
               6    -2.622245
```

```
    7   -2.603370
    8   -2.585140
    9   -2.583143
   10   -2.579886
   11   -2.563037
   12   -2.553304
   13   -2.549820
   14   -2.521095
   15   -2.514894
   16   -2.509215
   17   -2.488150
   18   -2.482372
   19   -2.474471
 ---------------------------------------
  MSG:checking the probality of each eigenstates
All k-point calculation, which you requested,are done.


 ********************************
 *    Naniwa code is terminated *
 ********************************

 ----------------------------------------
```

44.2.5. EIGEN

It gives you the information about quantum number and eigenenergies (eV)

```
    0   -2.738340
    1   -2.705202
    2   -2.647118
    3   -2.643600
    4   -2.624484
    5   -2.624043
    6   -2.622192
    7   -2.603305
    8   -2.585474
    9   -2.583032
   10   -2.579732
   11   -2.562970
    ~        ~
    ~          ~
 2828    1.397943
 2829    1.401005
 2830    1.405767
 2831    1.427601
 2832    1.433854
 2833    1.451622
 2834    1.454133
```

4.2.6 stateXXXXX

It gives you the information about the XXXXXth eigenstate.

```
stateXXXXX
              0
   -2.73834024018558
    2.82029500000000          0.000000000000000E+000   0.000000000000000E+000
    1.41014750000000          2.44244710549300         0.000000000000000E+000
  0.000000000000000E+000   0.000000000000000E+000     10.7499987274312
              9             9             35
    10.1834736495052          10.1834736495052          10.6003270094309
              0             0             0
  (-1.777456224903666E-006,-4.974294848068331E-007)
              1             0             0
  (7.765809220410138E-007,4.430627433556088E-007)
              2             0             0
  (1.439251475885052E-006,-2.545852331603858E-007)
             ~             ~             ~

             ~             ~             ~
 (-5.370729930410416E-006,5.895488939631636E-007)
              5             8             34
  (-5.362392471165102E-006,2.837122256840745E-007)
              6             8             34
  (3.192389365300413E-006,-2.070464429167339E-007)
              7             8             34
  (-2.748703204133705E-006,1.452834001927643E-007)
              8             8             34
  (2.221894867527162E-006,0.000000000000000E+000)
```

state file name

quantum number, ω

eigenenergies (eV)

$\left(\overrightarrow{La_1}\right)_x$ $\left(\overrightarrow{La_1}\right)_y$ $\left(\overrightarrow{La_1}\right)_z$ (Å)

$\left(\overrightarrow{La_2}\right)_x$ $\left(\overrightarrow{La_2}\right)_y$ $\left(\overrightarrow{La_2}\right)_z$ (Å)

$\left(\overrightarrow{La_3}\right)_x$ $\left(\overrightarrow{La_3}\right)_y$ $\left(\overrightarrow{La_3}\right)_z$ (Å)

n_{g1} n_{g2} n_{g3} : Numbers of Gaussian grid alog $\overrightarrow{a_1}$, $\overrightarrow{a_2}$, $\overrightarrow{a_3}$, where $n_{g1} \times n_{g2} \times n_{g3} = N_G$

σ_x σ_y σ_z : Gaussian decay factors

i_{g1}^1 i_{g2}^1 i_{g3}^1 : 1st grid point indicators alog $\overrightarrow{a_1}$, $\overrightarrow{a_2}$, $\overrightarrow{a_3}$.

$\left(\mathrm{Re}\, C_{\omega,1},\ \mathrm{Im}\, C_{\omega,1}\right)$: corresponding coefficient of eigenvector

i_{g1}^2 i_{g2}^2 i_{g3}^2 :2nd grid point indicators alog $\overrightarrow{a_1}$, $\overrightarrow{a_2}$, $\overrightarrow{a_3}$.

$\left(\mathrm{Re}\, C_{\omega,2},\ \mathrm{Im}\, C_{\omega,2}\right)$: corresponding coefficient of eigenvector

⋮

$i_{g1}^{N_G}$ $i_{g2}^{N_G}$ $i_{g3}^{N_G}$:N_G-th grid point indicators alog $\overrightarrow{a_1}$, $\overrightarrow{a_2}$, $\overrightarrow{a_3}$.

$\left(\mathrm{Re}\, C_{\omega,N_G},\ \mathrm{Im}\, C_{\omega,N_G}\right)$: corresponding coefficient of eigenvector

5. How to use (minimum)

5.1 Making 3D-potential energy surface for quantum particle in material.

You must make the potential energy surface for your quantum particle motion in materials by ab initio electron state calculation.

Calculate the total energy : $E(X,Y,Z)$ of electron system as a function of quantum particle position (X,Y,Z). Use it as adiabatic potential energy: $U(X,Y,Z)$.

Save these potential energy information into POTENG file. (See the section 4.2 files section)

5.2 Making the simulation setting

Make the setting file: INSET (See the section 4.2 files section).

Copy the input file: INSET, POTENG on your working directory.

5.3 Executing the simulation program.

Type following command line on your working directory.

~/naniwa/bin/naniwaSS_run [Enter]

If successful, you can obtain the LOG file and the kpoint0000 directory.

Please check the contents in the LOG files.

You can find the simulation results under the kpoint0000 directory.

If "Oracle Grid Engine" is installed on your computer, you can submit your simulation by following command line:

qsub run.csh [Enter]

qstat [Enter]

5.4 Check simulation results

Move to sub-directory: "kpoint0000"

cd ./kpoint0000 [Enter]

ls [Enter]

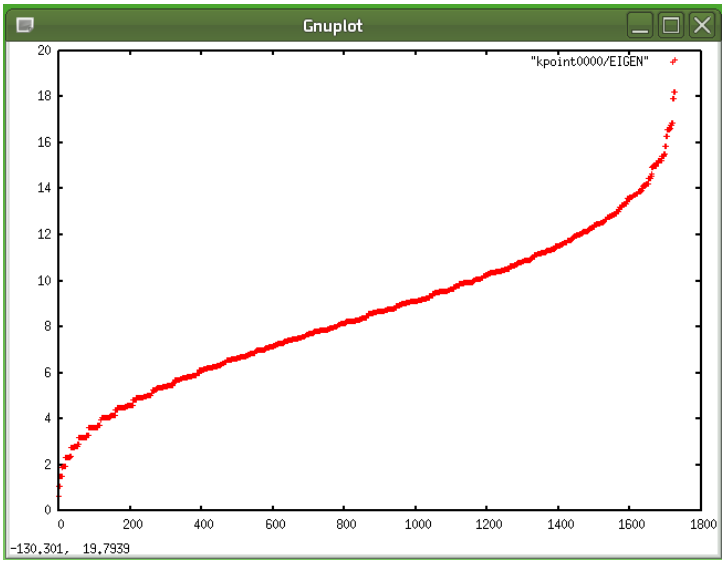
cat EIGEN [Enter]

5.4.1 How to see the Eigenenergies as a function of quantum number.

gnuplot [Enter]

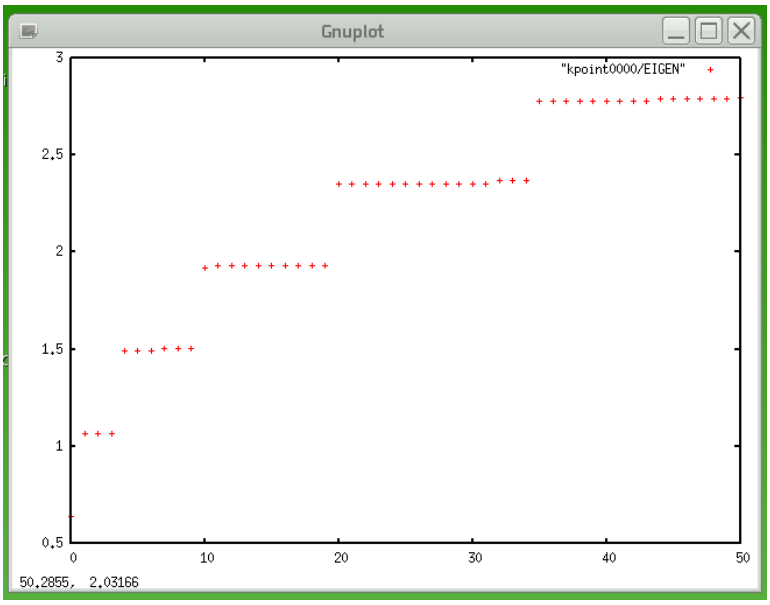
```
G N U P L O T
        Version 4.0 patchlevel 0

        ....

        ....

        Send bugs, suggestions and mods to
                <gnuplot-bugs@lists.sourceforge.net>


Terminal type set to 'x11'
```

gnuplot> *plot "EIGEN" [Enter]*



gnuplot> *set xrange [0:50]* *[Enter]*

gnuplot> *replot* *[Enter]*



Quit the gnuplot

gnuplot> *quit* *[Enter]*

5.4.2 How to see the Wave functions

5.4.2.1 OpenDX case

Naniwa package includes the “WaveViewer3D”, which is Visual Program running on the OpenDX.

(1) Convert the eigenstate data for OpenDX

Make the OpenDX data from the groud state to 10^{th} excited state.

cp ../INSET ./ [Enter]

../../../bin/state2dx 0 10 [Enter]

```
stat2dx
 I create the files for Opendx from                0 to               10
 -------------------------
 I got the following data from INSET file.
 (n1,n2,n3)=                9              9              9
 I set (n1,n2,n3)=             27             27             27
 -------------------------
 ================================
  POTDAT.cfg is not found.
  Position origine is reset to
     0.0000000E+00   0.0000000E+00   0.0000000E+00
 ================================
 wave00000.dat   was created.
 WAVE function
     Max value=     2.219193
     Min value= -4.8887818E-03
  probability=   0.9999163

 wave00000.general   was created.gnuplot>
     …
wave00010.dat   was created.
 WAVE function
     Max value=     1.522613
     Min value=    -1.618646
  probability=   0.9998179

 wave00010.general   was created.
```

(2) Convert the potential energy surface (PES) data for OpenDX

cp ../POTENG ./ [Enter]

../../../bin/poteng2dx [Enter]

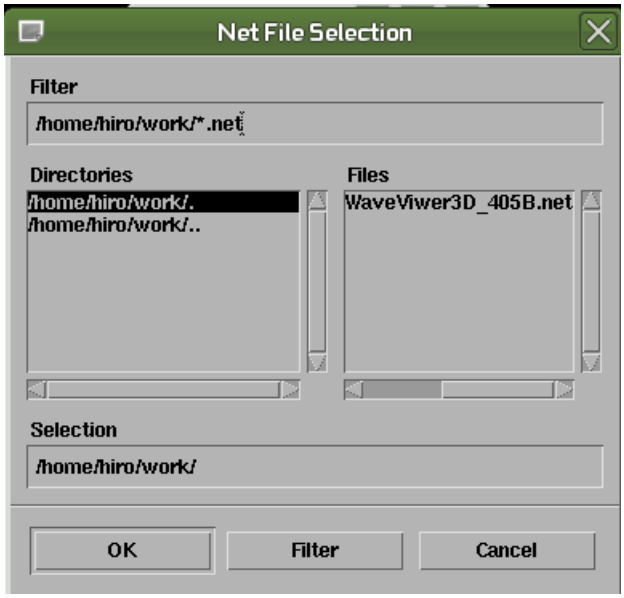
```
================================
  POTDAT.cfg is not found.
  Position origine is reset to
     0.0000000E+00   0.0000000E+00   0.0000000E+00
 ================================
 "PES3D.dat" file was created.
 Umax=     1.393875
 Umin=    0.0000000E+00
 "PES3D.general" file was created.
```

(3) Execute the OpenDX

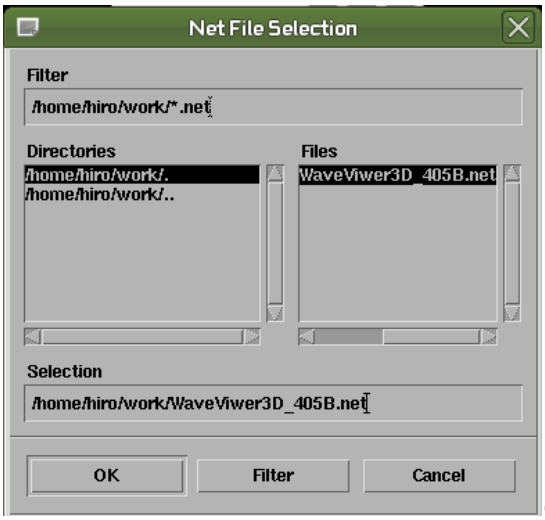
cp ../../../etc/OpenDX/WaveViwer3D_405B ./ [Enter]*

dx [Enter]

Click “Run Visual Programs”



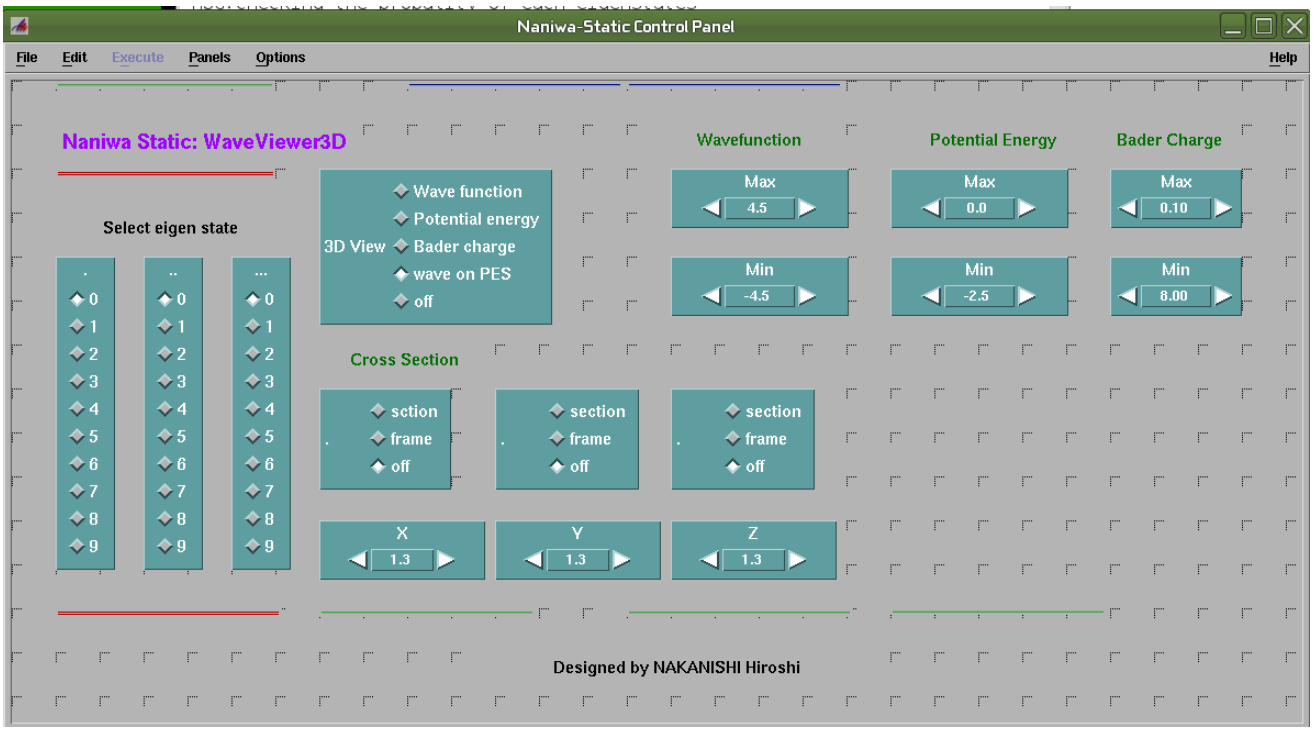
(4) Select the “WaveViewer3D_405B.net in right hand winodw.



Click “OK” button.

(5) Operate the “WaveViewer3D”,

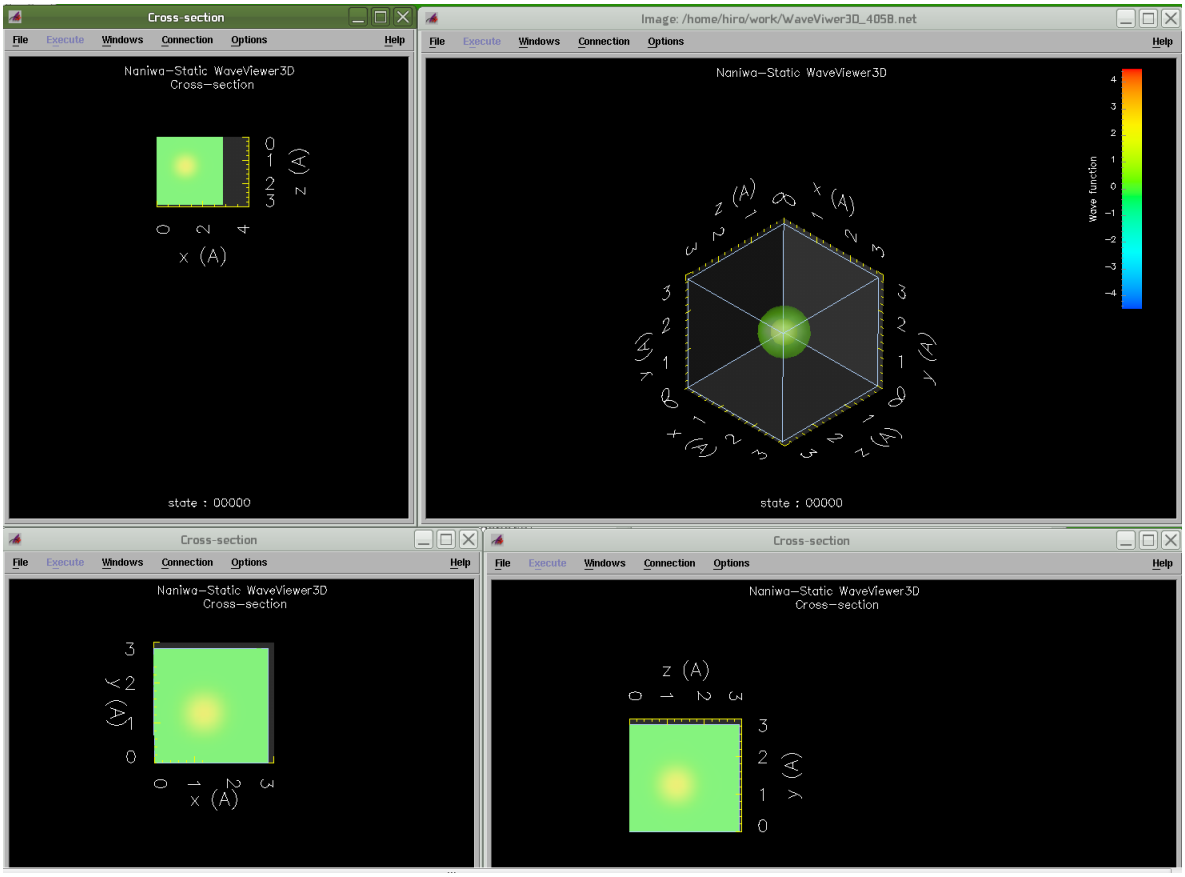
You can control it from Control panel.



(6) See the visualized results

You can see the 3-dimensional model in 4 Visualization windows;

top view, bird view, side view and bottom view.



(7) Quit the OpenDX

Click “Quit” button.

5.4.2.2 XcrySDen case

Naniwa package includes the Output file converter for XCrySDen.

(1) Convert the eigenstate data for XCrySDen.

Make the XCrySDen data from the groud state to 10th excited state.

cd ~/naniwa/work/test/kpoint0000 *[Enter]*

../../../bin/state2xsf 0 10 *[Enter]*

```
stat2xsf
 I create the files for XCrySDen from                0 to               10
 --------------------------
 ERROR no INSET file :reset
 n1 = 30
 n2 = 30
 n3 = 90
 ------------------------
 ================================
  POTDAT.cfg is not found.
  Position origine is reset to
    0.0000000E+00   0.0000000E+00   0.0000000E+00
 ================================
 wave00000.xsf   was created.
 WAVE function
    Max value=    2.18718407153673
    Min value= -4.841641351593371E-003
  probability=   0.999931342158014


 wave00001.xsf   was created.
 WAVE function
    Max value=    2.02312748202589
    Min value=   -1.86989237272753
  probability=   0.999900567109712


 wave00002.xsf   was created.
 WAVE function
    Max value=    1.94090713382515
    Min value=   -1.94489431345728
  probability=   0.999900392093716
```

(2) Convert the potential enrgy surface (PES) data for XCrySDen

cp ../POTENG ./ *[Enter]*

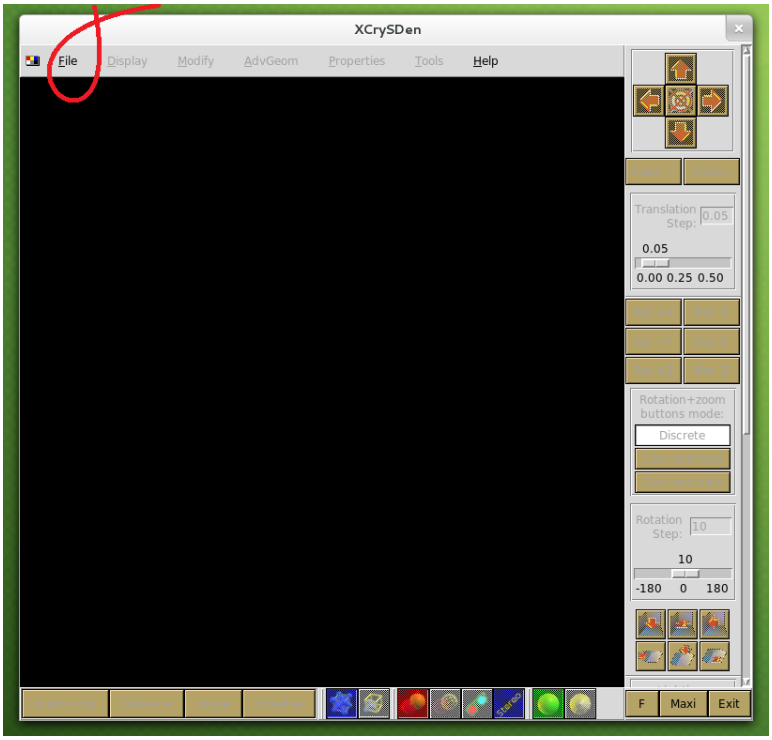
cp ../INSET ./ [Enter]

../../../bin/poteng2xsf *[Enter]*

```
 **********************************

   PES data   converter for XCrySDen

  **********************************

  "PES3D.xsf" file was created.

  Umax=    1.393875

  Umin=   0.0000000E+00
```

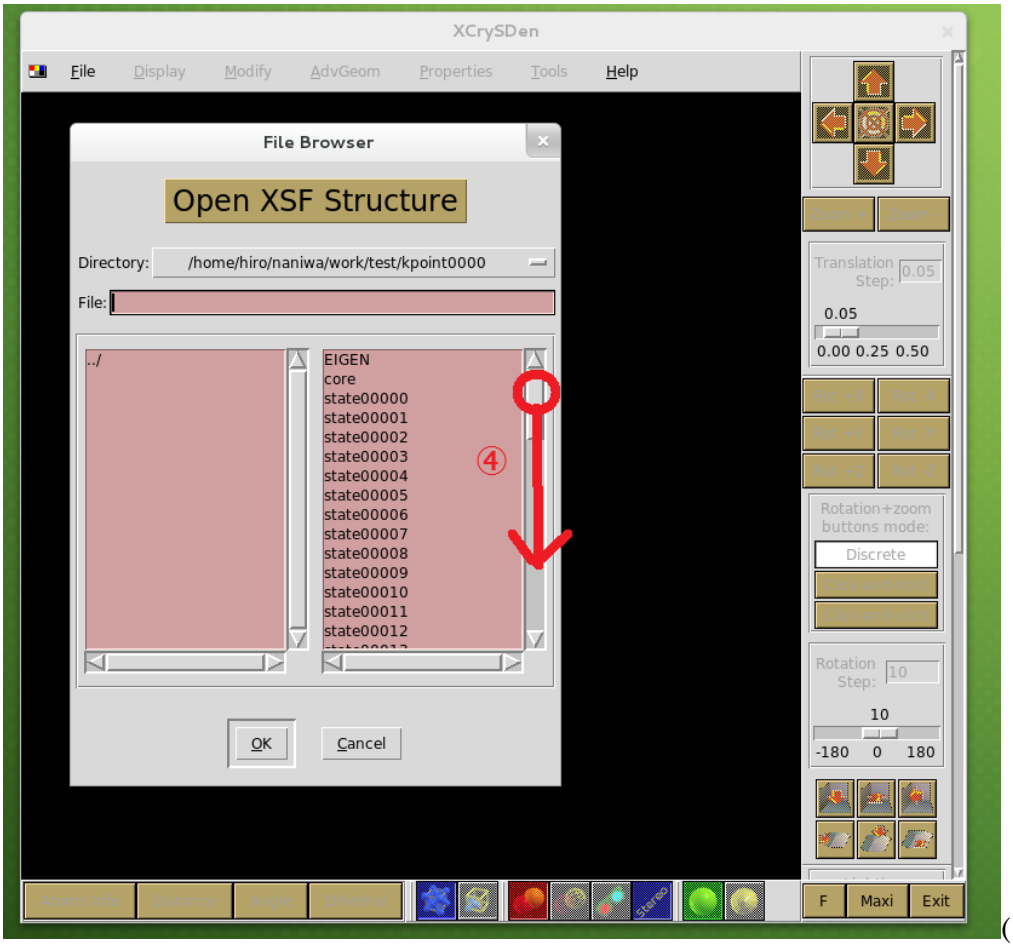
(3) Execute the XCrySDen

xcrysden *[Enter]*

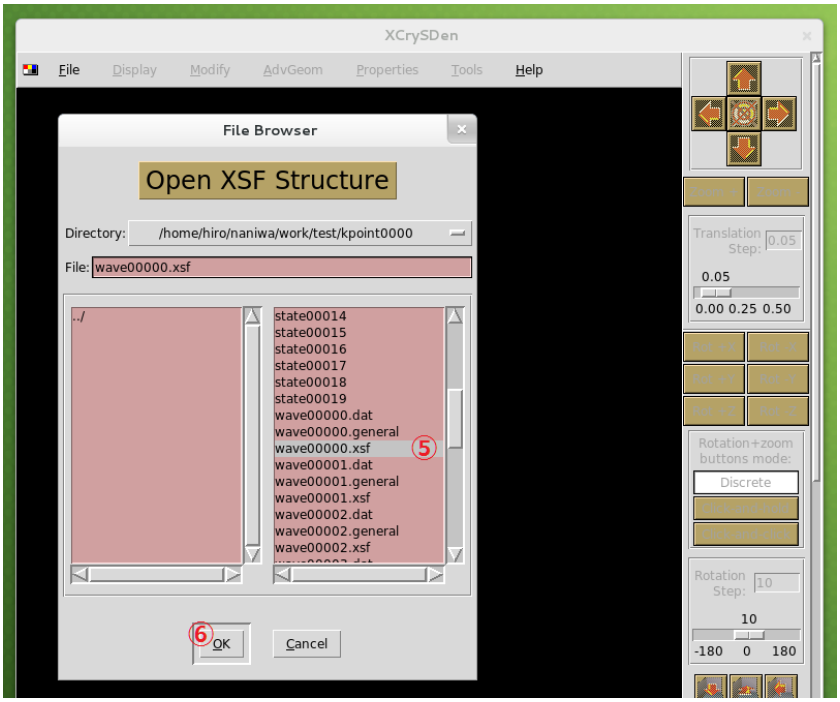


(4) Open the file wave00000.xsf

① Click "File" ② Click "Open Structure" ③ Click "Open XSF (XCrySDen Structure file)"



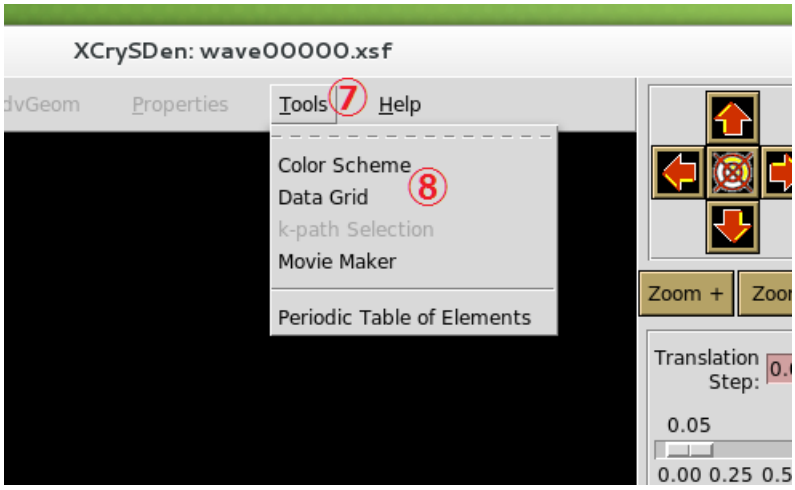
④Drag down the thumb in 2nd vertical scrollbar.



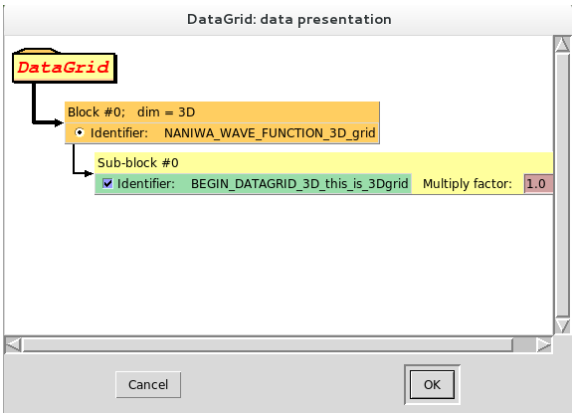
⑤Select the file “wave00000.xsf”

⑥Click “OK”

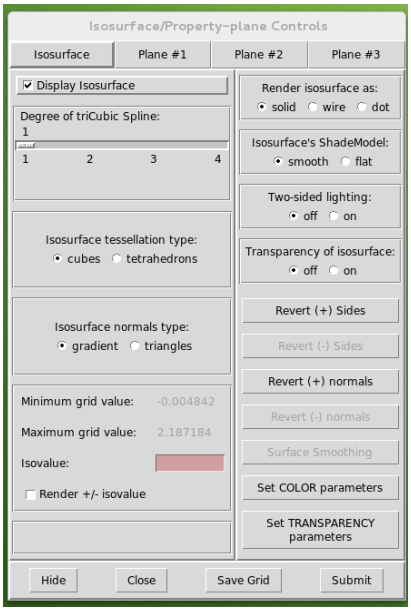
5) Set the view status.



⑦Click “Tools”　⑧Click “Data Grid”

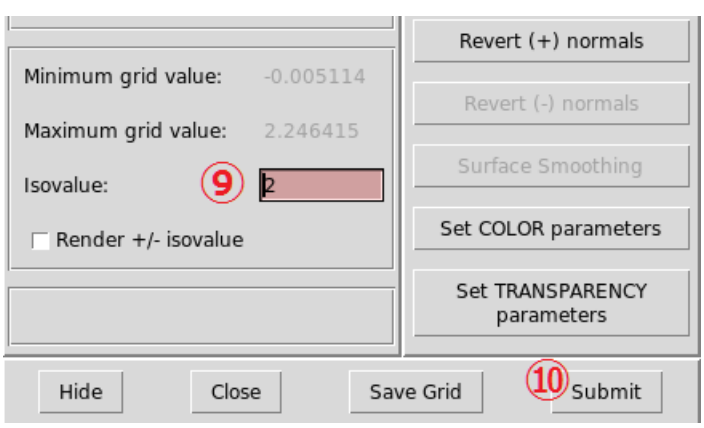


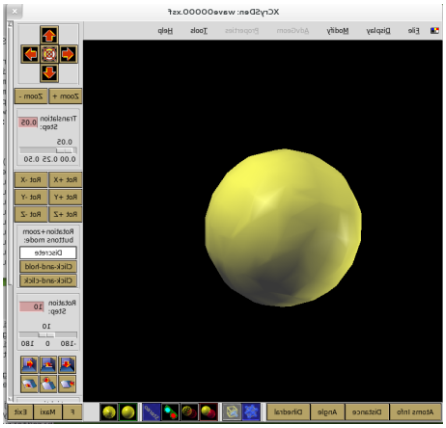
Click ”OK” button.



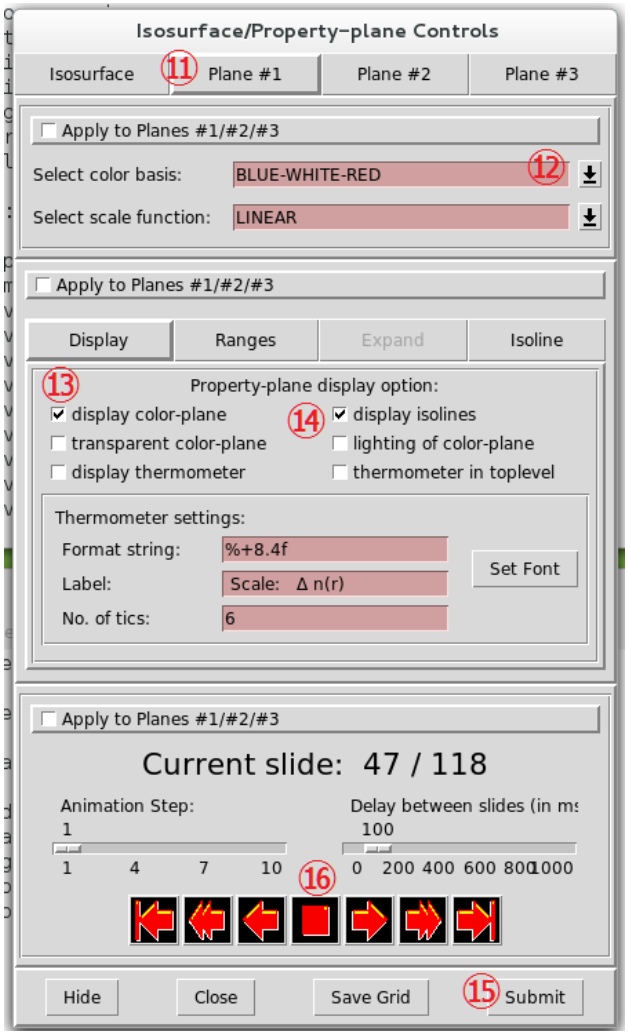
Then, you can see this “Isosurface/Property-plane Controls”

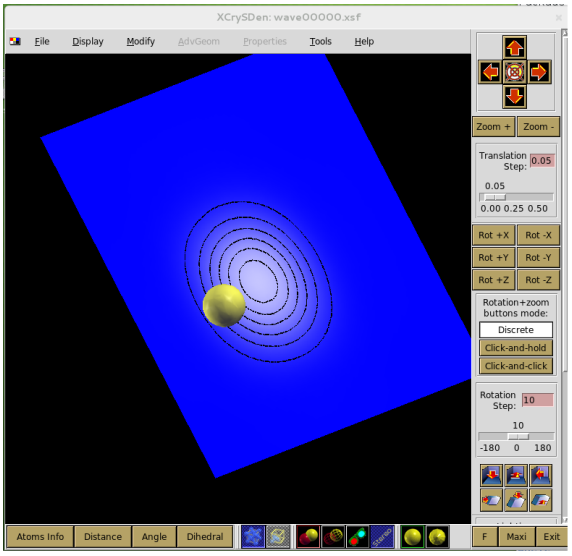
(6)Set adequate Isovalue ⑨ and click “Submit” botton ⑩. You can see its isosurface,

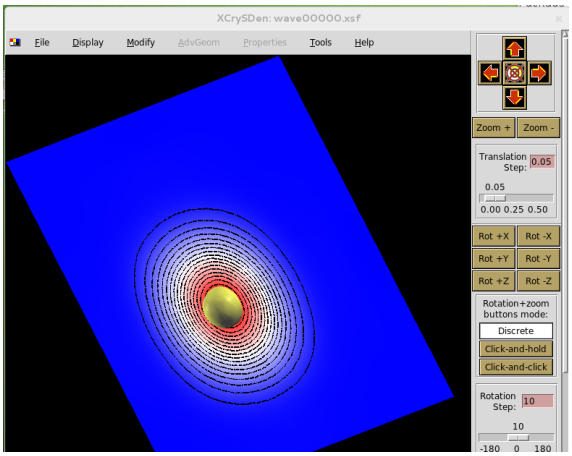




(7) Select “Plane #1” ⑪ and choose select color basis ⑫ . And check boxes of display color-plane⑬ and display isolines ⑭. Click submit botton ⑮ and you can see the isoline on a cross section, which can be moved by arrows ⑯.

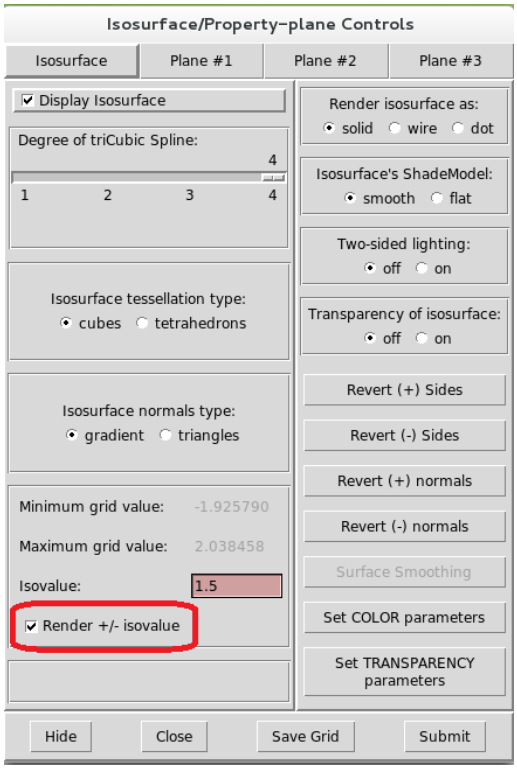


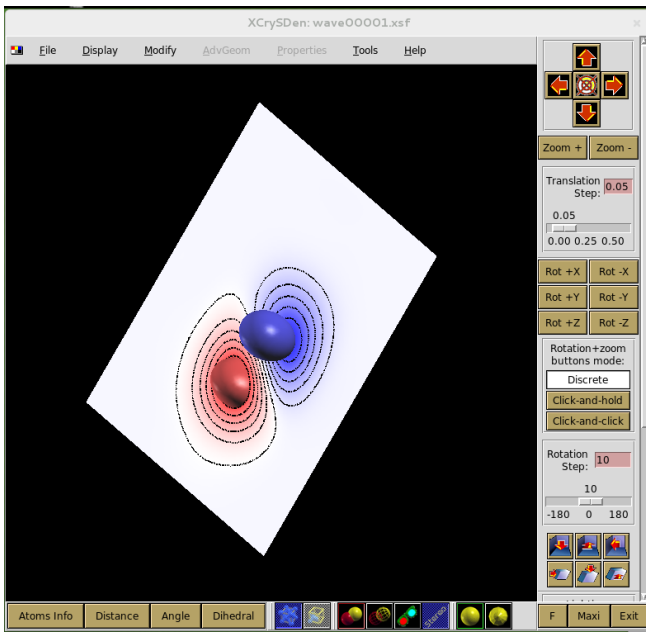




“wave00000.xsf” case

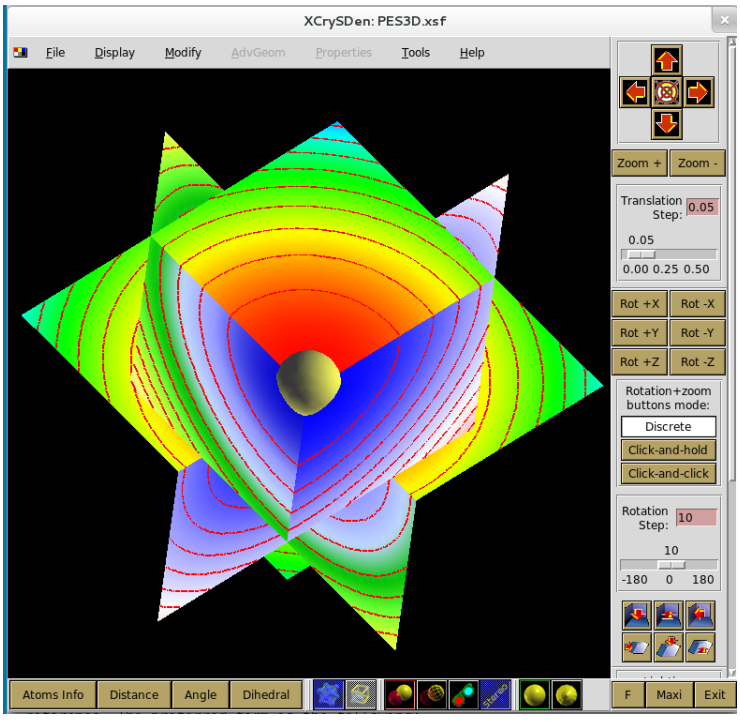
(8) You can also see the excited states whose wave function has node.





"wave00001.xsf" case

(6) You can also see the potential energy surface



"PES3D.xsf" case

(7) Quit the XCrySDen

Click "File" and "Exit"

And click "Yes"

5.4.2.3 VESTA case

VESTA can read xsf files to show the wavefunctions

(1) Make xsf files.

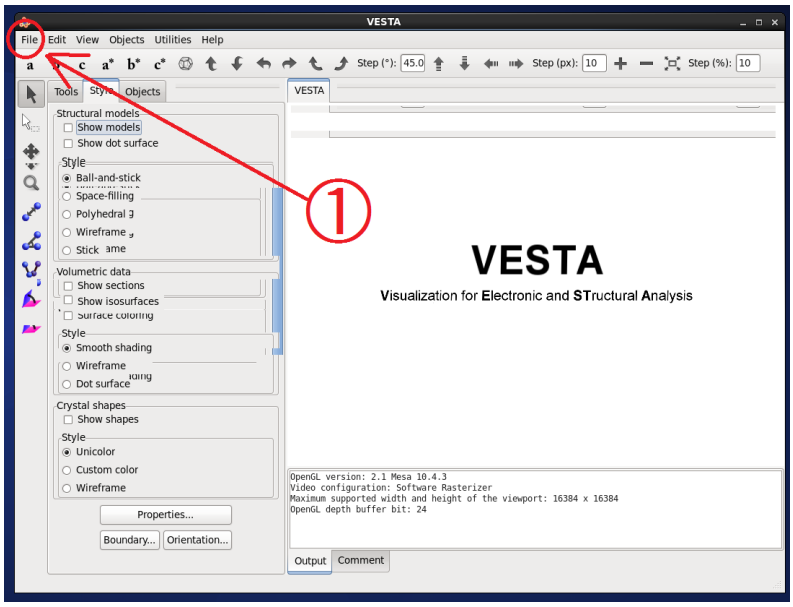
Do the same procedures from (1) to (2) in section 5.4.2.2.

(2) Execute the VESTA

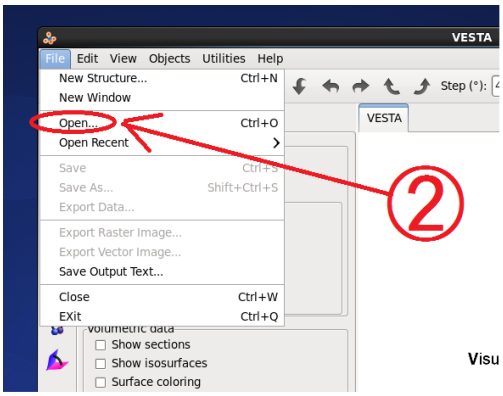
VESTA [Enter]

(3) Open the file wave00000.xsf

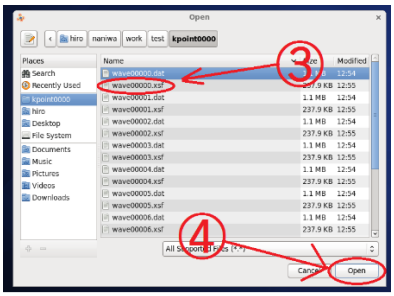
① Click “File”



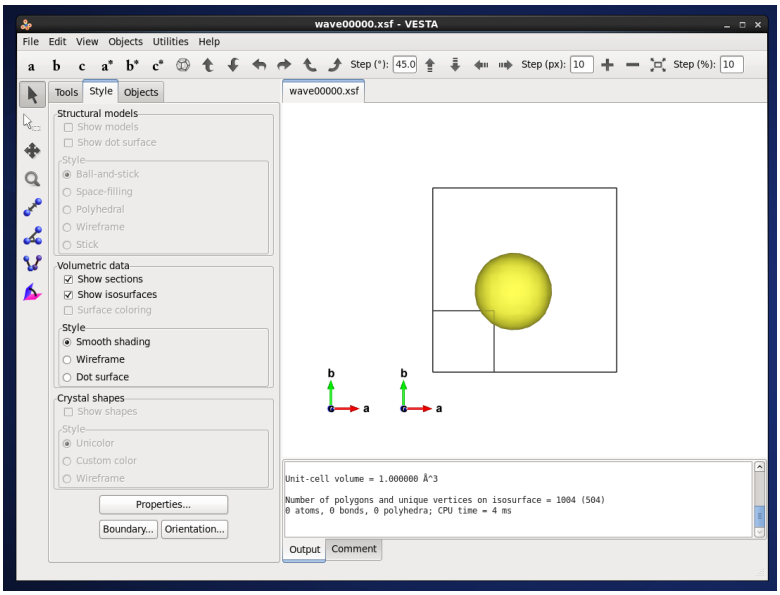
② Click “Open”



③ Select the file “wave00000.xsf”. ④Click “OK”

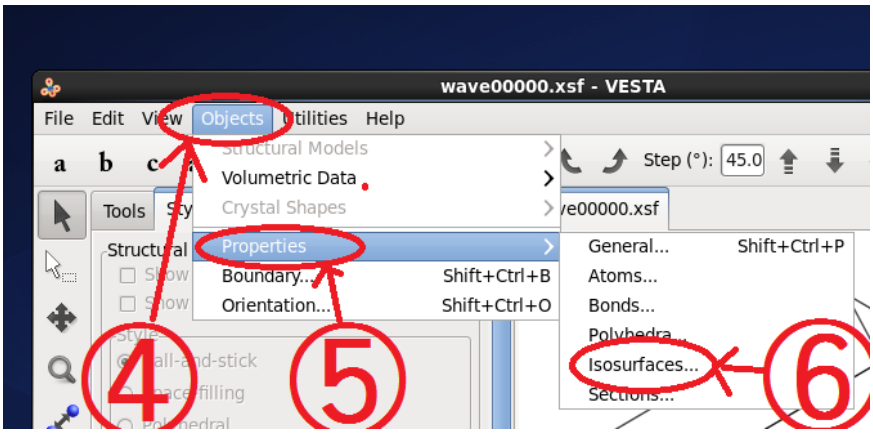


(4) You can see the ground state wave function.

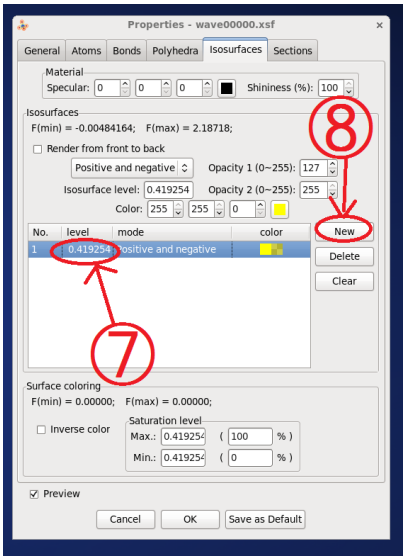


(5) You can control the setting of isosurface.

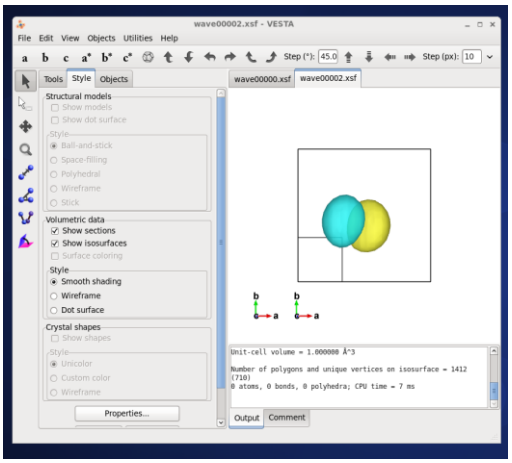
④ Click "Object" ⑤ Select "Property" ⑥ Click "Isosurface…"



⑦You can change the level of isosurface and ⑧ add the other levels.



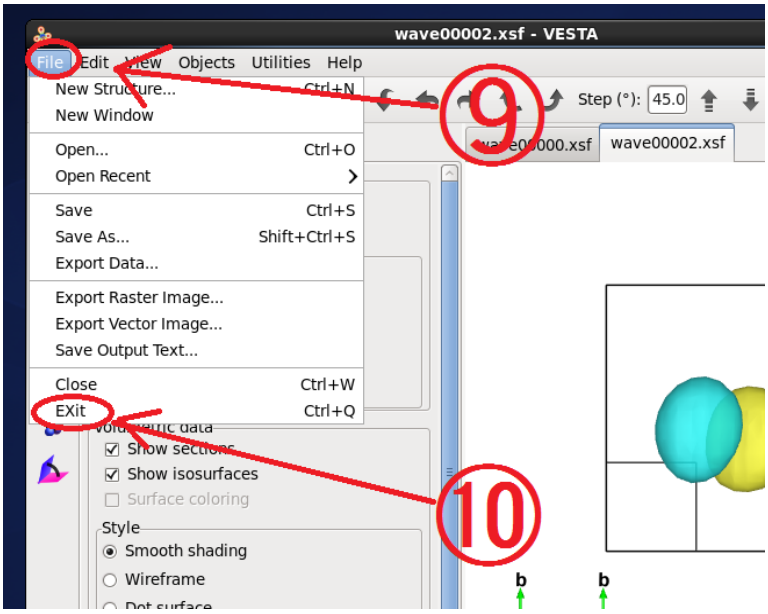
(6) You can also see the excited states whose wave function has node.



“wave00001.xsf” case

(7) Quit the VESTA

⑨ Click “File” and ⑩ “Exit”



References

[1] Hideaki Kasai and Ayao Okiji, Progress in Surface Science, Vol. 44, No. 2, pp. 101-183 (1993).

[2] Wilson Agerico Diño, Hideaki Kasai and Ayao Okiji, Progress in Surface Science, Vol. 63, No. 3-5, pp. 63-134 (2000).

[3] Japanese Patent No. 4774523.

[4] Japanese patent application 2012-16133.

[5] Kunihiro Nobuhara, Hiroshi Nakanishi, Hideaki Kasai and Ayao Okiji, Surface Science, Vol. 493, No. 1-3, pp. 271-277 (2001).

[6] Kunihiro Nobuhara, Hiroshi Nakanishi, Hideaki Kasai and Ayao Okiji, Surface Science, Vol. 493, No. 1-3, pp. 271-277 (2001).

[7] Kunihiro Nobuhara, Hideaki Kasai, Hiroshi Nakanishi and Ayao Okiji, Surface Science, Vol. 507-510, pp. 82-86 (2002).

[8] Kunihiro Nobuhara, Hideaki Kasai, Hiroshi Nakanishi and Ayao Okiji, Journal of Applied Physics Vol. 92, No. 10, pp. 5704-5706 (2002).

[9] Kunihiro Nobuhara, Hideaki Kasai, Hiroshi Nakanishi and Wilson Agerico Diño, Journal of Applied Physics Vol. 96, No. 9, pp. 5020-5025 (2004).

[10] Nobuki Ozawa, Tanglaw Abat Roman, Hiroshi Nakanishi, Hideaki Kasai, Surface Science, Vol. 600, No. 18, pp. 3550-3554 (2006).

[11] Tanglaw Roman, Wilson Agerico Diño, Hiroshi Nakanishi, Hideaki Kasai, e-Journal of Surface Science and Nanotechnology, Vol. 4, pp. 619-623 (2006).

[12] Nobuki Ozawa, Tanglaw Roman, Hiroshi Nakanishi, Wilson Agerico Diño, Hideaki Kasai ,Physical Review B Vol.75, No.11, pp. 115421 (2007).

[13] Nobuki Ozawa, Tanglaw Roman, Nelson Buntimil Arboleda Jr., Wilson Agerico Diño, Hiroshi Nakanishi, Hideaki Kasai, Journal of Physics: Condensed Matter, Vol. 19, No. 48, pp. 365214 (2007).

[14] Nobuki Ozawa, Nelson Buntimil Arboleda Jr., Hiroshi Nakanishi, Hideaki Kasai, Surface Science, Vol. 602, No. 4, pp. 859-863 (2008).

[15] Nobuki Ozawa, Nelson Buntimil Arboleda Jr., Hiroshi Nakanishi, Nobuaki, Shimoji, Hideaki Kasai, Surface and Interface Analysis, Vol. 40, Issue 6-7, pp. 1108-1112 (2008).

[16] https://software.intel.com/en-us/intel-parallel-studio-xe

[17] https://software.intel.com/en-us/intel-mkl
Intel-MKL is included in Intel-Composer-XE [16]

[18] http://www.gnuplot.info/

[19] http://www.opendx.org/ (dead link)

[20] http://www.xcrysden.org/

[21] http://jp-minerals.org/vesta/en/

Appendix

A Porting

Naniwa standard version uses the following system-dependent subroutines.

(1) Intel Fortran built-in subroutine: SYSTEM

(2) Lapack subroutine: ZHEGV

(3) Intel MKL procedure: MKL_DFTI

If you want to port Nanwia to other system, you must replace these routines. Please rewrite the (1) UNIXcommand_intel.f, (2) diagonalize_lapack.f and (3) fastDFT3D_MKL.f in SRC directory.

And also you must replace Fortran compiler name and its options.

These are defined in Makefile

```
#*        for intel FORTRAN & MKL
#*
#
ASR =   diagonalize_lapack.f    UNIXcommand_intel.f   fastDFT3D_MKL.f
LIB = /opt/intel/mkl/lib/intel64
INC = -I/opt/intel/mkl/include

#*Compiler name and option
FC = ifort
OPT = -O3 -mkl -lpthread
```

A-1. Porting to Hitachi SR16000 with optimized FORTRAN90, MATRIX/MPP and LAPACK

We provide the necessary interface to Hitachi SR16000. You only have to modify the following parts in Makefile.

Comment out part:

```
#*        for intel FORTRAN & MKL
#*
#
#ASR =   diagonalize_lapack.f    UNIXcommand_intel.f   fastDFT3D_MKL.f
#LIB = /opt/intel/mkl/lib/intel64
#INC = -I/opt/intel/mkl/include

#*Compiler name and option
#FC = ifort
#OPT = -O3 -mkl -lpthread
```

Please insert mentioned-above red # in Makefile.

Uncommented part:

```
#*        for Hitachi Optimizing FORTRAN90
#*                  LAPACK    & MATRIX/MPP
#*
#*                           KEK SYSTEM A
#*
#ASR = diagonalize_lapack.f UNIXcommand_hitachi.f fastDFT3D_MPP.f
#LIB = /opt/hitachi/matmpp/lib -lmatmpp -L/srhome/.tools/lib -llapack_sc -lblas_sc
#INC =

#*Compiler name and option
#FC = f90
#OPT = -Os -64 -model=M1 -parallel
```

Please remove the mentioned-above blue # in Makefile.

These procedure replaces SYSTEM and MKL_DFTI to HF_SH and HZFT6M. Later subroutine restricts the value of n_1, n_2, n_3 in POTENG. These integer values must be not less than 8 and power of 2. Please see the naniwa/etc/example/testSR.

B Making 3D-potential energy surface by VASP

example: H in Si bulk

See the directory: ~/naniwa/etc/potentials/SiBulk

under ~/naniwa/etc/potentials/SiBulk

host : getting the total energy of host material without hydrogen atom

particle: getting the total energy of hydrogen atom without host material

scan: getting the total energy in the case of hydrogen atom in host material as a function of hydrogen atom position

(1) **cd** ~/naniwa/etc/potentials/SiBulk/host/

qsub run.csh

After the calculation, getting the energy E_host from last Eo value in OSZCAR

(2) **cd** ~/naniwa/etc/potentials/SiBulk/particle/

qsub run.csh

After the calculation, getting the energy E_partcile from last Eo value in OSZCAR

(3) **cd** ~/naniwa/etc/potentials/SiBulk/scan/

csh ./auto.csh

After all calculations, ..

csh ./collect.csh

cp POTDAT ./ POTDAT2POTENG

cd ./ POTDAT2POTENG

ifort makePOTENG.f -o makePOTENG

vi POTDAT.cfg

Edit the configuration file: POTDAT.cfg

input the value of E_host and E_partcile into this file

```
*title:  Potential energy for H in Si bulk
*lattice information
 3.83958982184
 1.00000000000  0.0                0.0
 0.50000000000  0.86602540378  0.0
 0.50000000000  0.28867513459  0.81649658092
* data mesh numbers
 20  20  20
*origine of positioni (fractional)
 0.00000000 0.00000000 0.00000000
*isolated  particle energy (eV)
 -.11089857E+01
*isolated slab (host) energy (eV)
 -.86414775E+02
*cut of energy (eV)
 2.00
```

./makePOTENG

Then you can obtain the POTENG file

For checking the POTENG file.

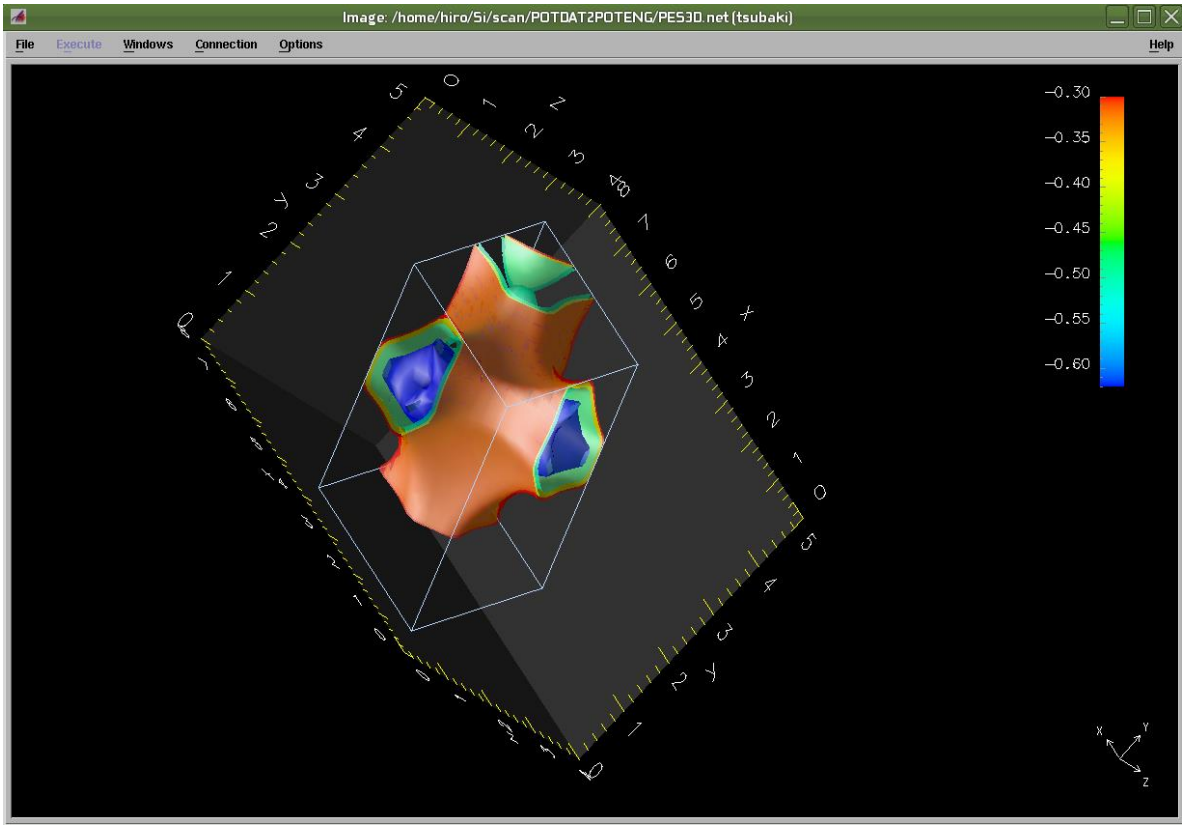
ifort POTENG2DX.f -o POTENG2DX

dx

Click “Run Visual Programs”

Select the “PES3D.net in right hand winodw.

Click “OK” button.

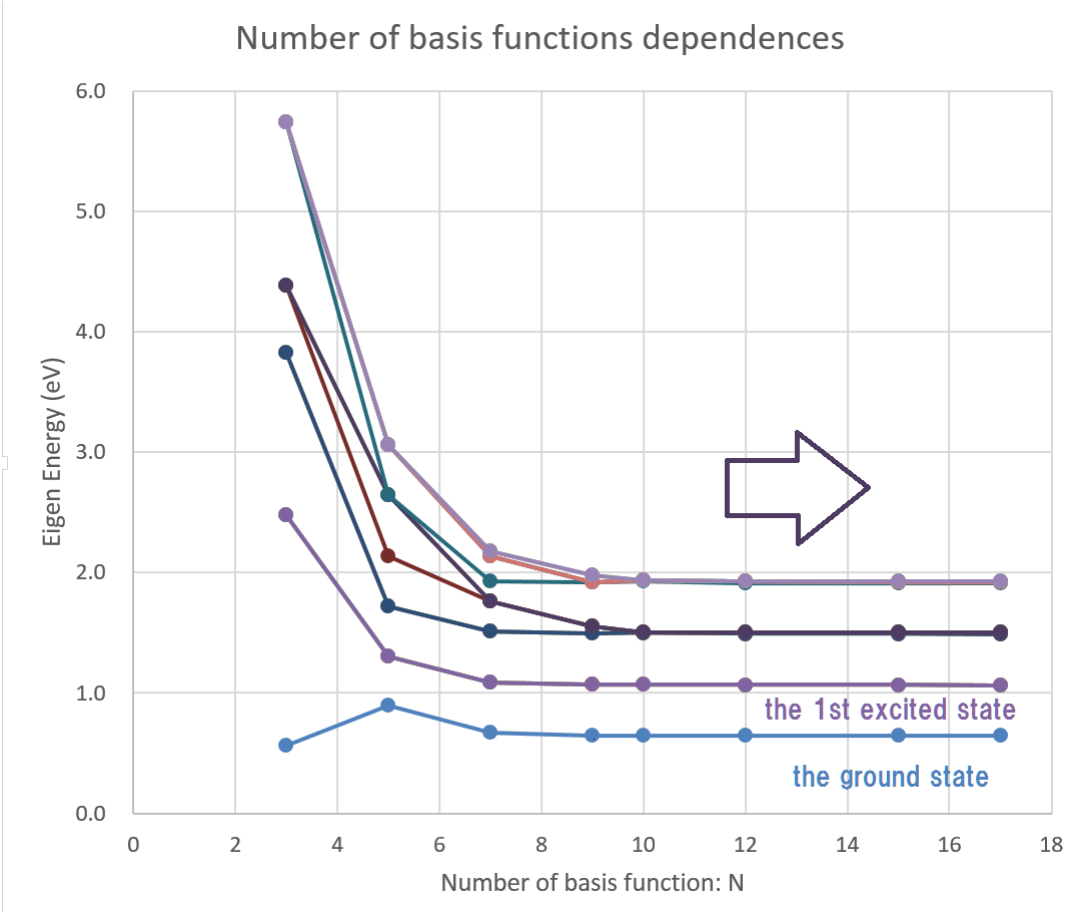


C　Required parameters for convergence

In order to get the reliable results, you should try the various parameters and check the convergence of their results.

C.1 Number of nodes for the 3D Gaussian functions

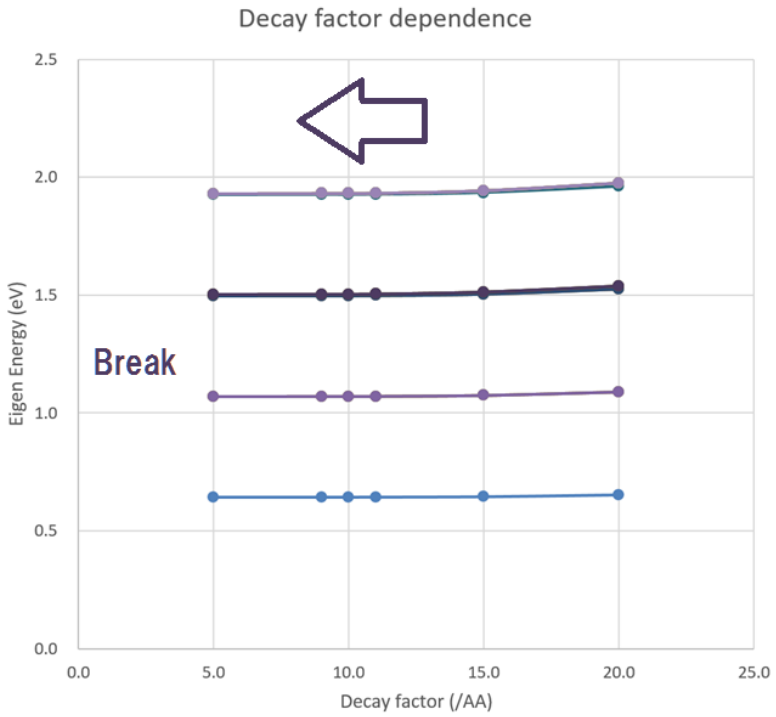
Here we show the Eigenenergy convergency as a function of the number of nodes for the 3D Gaussian functions. When only the number of the nodes are changed in the test program: naniwa/etc/example/Harmonic3D, you can see the following dependences in the figure.



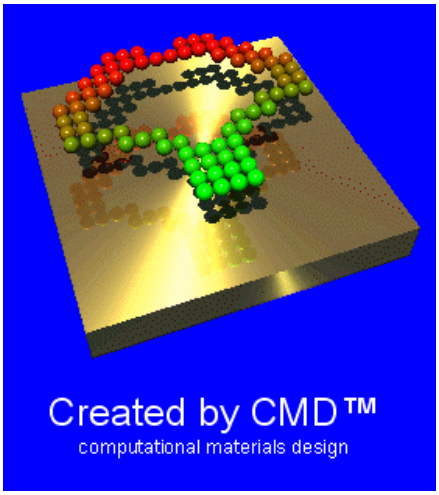
As the numbers of the grid increase, the eigen energies converge and you can get reliable results.

C.2 Gaussian decay factor (SigmaX, SigmaY SigmaZ)

Here we show the Eigenenergy convergency as a function of the decay factor of basis Gaussian functions. When only the decay factors are changed in the test program: naniwa/etc/example/Harmonic3D with fixed numbers of the grids: 10x10x10, you can see the following dependences in the figure.



As the decay factors decrease, the calculated Eigen energies converge, and you can get reliable results. But the decay factors decrease less than certain value, the calculations are suddenly broken. You must also check their wave function shapes.



Nakanishi Laboratory

National Institute of Technology, Akashi College, Japan
Phone & Facsimile: +81-78-946-6229
Lab. WEB: http://www.akashi.ac.jp/~nakanishi/
E-mail: nakanishi@akashi.ac.jp