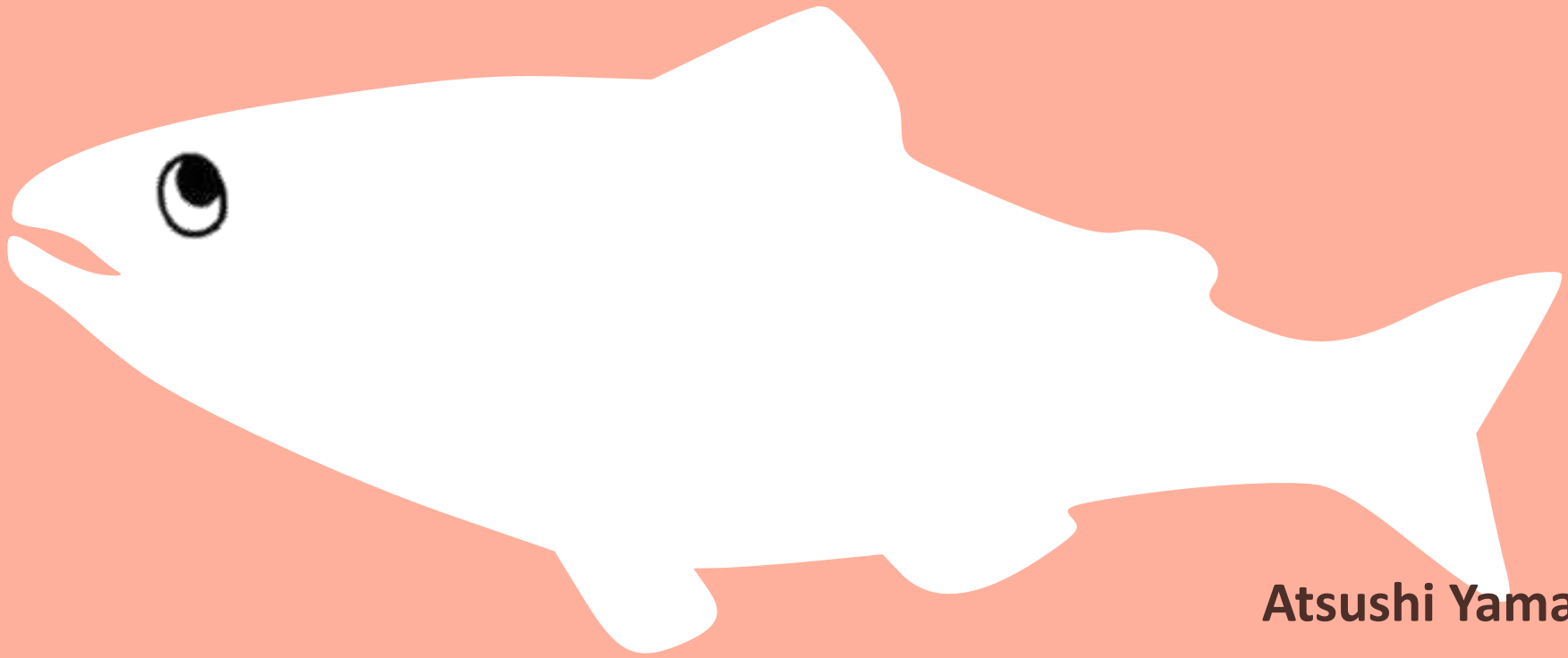
How to Use SALMON-2: Periodic Systems

Exercise



Atsushi Yamada

Center for Computational Sciences, University of Tsukuba

Basic Calculation Exercise(Periodic System) Introduction

Option for crystal system (periodic boundary condition):

“iperiodic = 3”

(Input file)

```
....
  sysname = 'Si'
/

&system
  iperiodic = 3
  al = 10.26d0,10.26d0,10.26d0
  isym = 1
  nstate = 32
....
```

Menu of the exercise

Calculations of silicon (Si) :

1. **Ground State**
2. **Dielectric Function**
3. **Real-Time TDDFT Simulation with Incident Laser Pulse**

1. Ground State Calculation (Job Script)

```
%> sbatch  ./job.sh
```

Put following input files in the same directory

- `Si_sc_gs.inp` (input file)
- `Si_rps.dat` (pseudo potential data file of Si)

```
#!/bin/bash
#SBATCH -J test
#SBATCH -p mixed
#SBATCH -N 4
#SBATCH --ntasks-per-node=2
#SBATCH --cpus-per-task=10
#SBATCH -t 0:30:00
#SBATCH -o stdout.log
#SBATCH -e stderr.log

export OMP_NUM_THREADS=$SLURM_CPUS_PER_TASK
module load intel intelmpi mkl
cd $SLURM_SUBMIT_DIR

input=./Si_sc_gs.inp
program=salmon.cpu

mpirun -np $SLURM_NTASKS $program < $input > out.log
```

Check parallelization and calculation time etc

Write input file name

Check path of program file

Write output file name for standard log

1. Ground State Calculation (input file)

Atomic unit [a.u.] is used as default.
To use Å/fs/eV unit system, give

```
&units
  unit_system='A_eV_fs'
/
```

(but W/cm^2 for electric field (laser) strength)

Input File: Si_sc_gs.inp

```
&calculation                          (Calculation Type)
  calc_mode = 'GS'                    <- It specifies ground state calculation only
/

&control                              (Control)
  sysname = 'Si'                      <- Body name of output file names. e.x. "Si_GS.out"(one of output file names)
/

&system                               (System)
  iperiodic = 3
  al = 10.26d0,10.26d0,10.26d0        <- Unit cell length (cuboid only)
  isym = 1                            <- no symmetry
  nstate = 32                         <- # of bands used in the calculation
  nelec = 32                          <- # of electrons
  nelem = 1                           <- # of atomic species
  natom = 8                           <- # of atoms
/                                     (usually, the same # of occupied and unoccupied orbitals can be used)

&pseudo                               (Pseudopotential)
  iZatom(1)=14                        <- Atomic number of the 1-th atomic species
  pseudo_file(1) = './Si_rps.dat'     <- File name of pseudopotential
  Lloc_ps(1)=2                        <- Number of the Reference angular momentum
/                                     (given in the pseudo-potential data file)

&functional                           (Functional)
  xc ='BJ_PW'                         <- Functional
/

&rgrid                                (space grid)   (multiples of 4 only)
  num_rgrid = 12,12,12                <- division number of unit cell length
/

&kgrid                                (# of k -points)   (Even number only. User specific k-points is available with reading option of external file)
  num_kgrid = 4,4,4
/

&scf                                  (Parameters of Ground State SCF)
  nscf = 120                          <- Maximum iteration number
  threshold=1.0d-7                    <- SCF convergence threshold: electron density difference per one electron
/

&atomic_red_coor                      (atomic coordinates)
     'Si'   .0    .0    .0    1
     'Si'   .25   .25   .25   1
     'Si'   .5    .0    .5    1
     'Si'   .0    .5    .5    1
     'Si'   .5    .5    .0    1
     'Si'   .75   .25   .75   1
     'Si'   .25   .75   .75   1
     'Si'   .75   .75   .25   1
/
```

[species, x,y,z, species-ID]

Here is reduced coordinate (unit cell length unit).
Use "&atomic_coor" for a.u.(or Å) unit,

1. Ground State Calculation (Job Submission, Output files)

Job submission on COMA

```
%> sbatch  ./job.sh
```

Check of jobs:

```
%> squeue –u (user name)
```

Job cancel :

```
%> scancel (JOBID)
```

List of Output Files:

- variables.log --- List of inputted variables(keywords) in the calculation
- out.log --- Standard output log (file name given in your job script(job.sh))
- Si_gs_info.data --- Information of GS SCF calculation etc
- Si_k.data --- k-points information
- Si_eigen.data --- Orbital eigen energy values in GS (→ band structure)
- gs_wfn_k/ --- GS wave function
  (This can be used as initial guess in the next calculation)
- PS_Si_KY_n.dat--- Temporary data of pseudopotential
- stdout.log --- Standard output of system (file name given in job.sh)
- stderr.log --- Standard error of system (file name given in job.sh)

1. Ground State Calculation (Check of results)

Following printed parts are useful to check the calculation results.

1. Convergence of SCF calculation (→next page)
   - Standard output (out.log).
   - Si_gs_info.out (this file is generated after SCF finished)
2. Band gap
   - Si_gs_info.out

     Band gaps are printed in the lines of ”Fundamental gap”(indirect transition) and ”BG between same k-point”(direct transition).

1. Ground State Calculation (Check of Convergence of SCF)

out.log

```
.... (omit) .....
-------------------------------------------
 iter =       62
 Total Energy =   -31.2252432282157     -2.500217455292386E-007
    jav(1),jav(2),jav(3)=   0.551402E-12  -0.197001E-10   0.131539E-10
 (orbital eigen energies)
  1  -0.150535   2  -0.122854   3  -0.122854   4  -0.122854
  5  -0.027750   6  -0.027750   7  -0.027750   8   0.001569
  9   0.129800  10   0.129800  11   0.129800  12   0.151207
 13   0.151207  14   0.151207  15   0.168080  16   0.168080
 17   0.300588  18   0.304820  19   0.304820  20   0.304820
 21   0.338168  22   0.338168  23   0.349556  24   0.349556
 25   0.349556  26   0.360471  27   0.360471  28   0.360471
 29   0.487072  30   0.487079  31   0.487096  32   0.515053
 (forces on atoms)
     1    -0.000000    -0.000000     0.000000
     2    -0.000000    -0.000000    -0.000000
     3     0.000000     0.000000     0.000000
     4    -0.000000     0.000000     0.000000
     5     0.000000     0.000000    -0.000000
     6     0.000000     0.000000    -0.000000
     7    -0.000000    -0.000000    -0.000000
     8     0.000000     0.000000    -0.000000
 e_var_ave,e_var_max=  2.807887450831825E-009  6.335738233753371E-008
 diff-dns,ddns/nelec=  1.019596987160528E-007  7.949054588372419E-008
 -----------------------------------------------
 GS converged at  62
.... (omit) .....
```

Iteration number

Total energy, total energy difference from the last step

Eigen orbital energy of the first k-point

Forces on each atom

Average and Maximum of Eigen energy difference

Total electron density difference (square root of diff.),
Electron density difference per one electron
(absolute value of the diff.)

(The red one is used as default to judge convergence)

Convergence

(Information of GS convergence is also printed in Si_GS.out after the SCF finished)

2. Calculation of Dielectric Function (Input File)

1.Copy the pre-calculated ground state data

```
%> cp -r (pre-calculation directory)/gs_wfn_k/  ./
```

2.Input file for dielectric constant calculation: Si_sc_dielec.inp

Change the input file name in job script file.

Put in the same directory:
`Si_rps.dat` (Si pseudopotential data)

```
&calculation
  calc_mode = 'RT'
/

&control
  sysname = 'Si'
/
&system
  iperiodic = 3
  al = 10.26d0,10.26d0,10.26d0
  isym = 1
  nstate = 32
  nelec = 32
  nelem = 1
  natom = 8
/
&pseudo
  iZatom(1)=14
  pseudo_file(1) = './Si_rps.dat'
  Lloc_ps(1)=2
/

&functional
  xc ='BJ_PW'
/
&rgrid
  num_rgrid = 12,12,12
/
&kgrid
  num_kgrid = 4,4,4
/

&tgrid
 nt=3000
 dt=0.16
/

&propagation
  propagator='etrs'
/

&emfield
  trans_longi = 'tr'
  ae_shape1 = 'impulse'
  epdir_re1 = 0.,0.,1.
/

&analysis
 nenergy=1000
 de=0.001
/

&atomic_red_coor
      'Si'   .0     .0     .0     1
      'Si'   .25    .25    .25    1
     ..... (omit) ....
      /
```

Real-Time: i.e. RS-TDDFT

(time grid)

# of time steps

Time step size [a.u.]

(time integration)

Algorithm of time integration

(Laser/External field)

"Transverse"

Type of Laser (or electric field)

Direction of electric field vector

(analysis)

# of energy grid used for dielectric function

Energy grid size for dielectric function

2. Calculation of Dielectric Function (Output File)

List of Output Files:

- variables.log ---
- out.log ---

} Same as the example 1 (see example 1)

- Si_rt.data --- energies, vector potential, electric fields, current densities, etc, at each time step
- Si_lr.data --- data from linear response calculation including dielectric function ε(ω) (in the case of "impulse" keyword)
- Si_force.data --- forces on each atoms at each time step
- PS_Si_KY_n.dat ---
- stdout.log ---
- stderr.log ---

} Log from system (see example 1)

- sc_performance_20171027_155208.log --- measured calculation time

2. Calculation of Dielectric Function (Check of results)

- Output file format of Si_lr.data

  Column 1: Energy (au)
  Column 2-7: Re(σ_x), Re(σ_y), Re(σ_z), Im(σ_x), Im(σ_y), Im(σ_z),
  Column 8-13: Re(ε_x), Re(ε_y), Re(ε_z), Im(ε_x), Im(ε_y), Im(ε_z)

  Imaginary part of Dielectric function,
  which can be easily converted to absorption spectrum

```
%> gnuplot
```

```
gnuplot> p "Si_lr.data" u 1:13 w l
```

→Imaginary part of dielectric function (z-component) as a function of energy

or

```
gnuplot> p "Si_lr.data" u ($1*27.2114):13 w l
```

1au(Hartree)=27.2114eV

※Tips to get beautiful ε(ω):

- See K.Yabana, et.al., Phys.Rev.B 85,045134 (2012) for the treatment in the small region of ω.
- Use relatively short simulation time because smooth ε（ω）curve can not be obtained with long simulation time due to the discrete k-points.

3. Real-Time TDDFT Simulation with Incident Laser pulse (Input File)

(Although you can use option to read the ground state wave function as the last example, run of both of GS calculation & Real-Time TDDFT simulation is shown here.)

Input File: Si_sc_pulse.inp

Change the input file name in job script file.

Put in the same directory:
`Si_rps.dat` (Si pseudopotential data)

```
&calculation
  calc_mode = 'GS_RT'
/

&control
  sysname = 'Si'
/

&system
  iperiodic = 3
  al = 10.26d0,10.26d0,10.26d0
  isym = 1
  nstate = 32
  nelec = 32
  nelem = 1
  natom = 8
/

&pseudo
  iZatom(1)=14
  pseudo_file(1) = './Si_rps.dat'
  Lloc_ps(1)=2
/
```

GS + RT
(i.e. Ground State calc.
+ Real-Time TDDFT)

```
&functional
  xc ='TBmBJ'
/

&rgrid
  num_rgrid = 12,12,12
/

&kgrid
  num_kgrid = 4,4,4
/

&tgrid
 nt=3000
 dt=0.16
/

&propagation
  propagator='etrs'
/
```

```
&scf
  nscf = 120
  threshold=1.0d-7
/

&emfield
  trans_longi = 'tr'
  ae_shape1 = 'Acos2'
  rlaser_int_wcm2_1 = 1d14
  pulse_tw1 = 441.195136248d0
  omega1 = 0.05696145187d0
  epdir_re1 = 0.,0.,1.
/

&atomic_red_coor
      'Si'   .0     .0     .0     1
      'Si'   .25    .25    .25    1
      ..... (omit) .....
/
```

Type of External Electric Field. Vector potential with $\cos^2$-type envelop

Strength of electric field [W/cm^2]

Time width of pulse

Frequency of pulse (photon energy)

3. Real-Time TDDFT Simulation with Incident Laser pulse (Input File)

Other analysis options are available.

(→ use &analysis block in input file)

For example:

- Print DOS → Use keyword like “out_dos=‘y’”
- Print projection analysis → Use keyword like “projection_option=‘gs’”

  (→ number of excited electrons and holes)
- Print electron density (cube format etc) → use keywords like

  out_dns_rt = 'y'

  out_dns_rt_step = 3000

  format3d = 'cube'

3. Real-Time TDDFT Simulation with Incident Laser pulse (Output File)

List of Output Files: (※Other output files can be generated with analysis options in input file)

- variables.log---
- out.log ---

} Same as the example 1,2 (see example 1,2)

- Si_gs_info.data
- Si_eigen.data
- Si_k.data

} Output for ground state calculation (see example 1)

- Si_rt.data
- Si_lr.data
- Si_force.out

} Output for RT-TDDFT calculation (see example 2)

- PS_Si_KY_n.dat
- stdout.log
- stderr.log
- sc_performance_20171027_155208.log

} Log from system (see example 1,2)

3. Real-Time TDDFT Simulation with Incident Laser pulse (Check of Results)

- Format of output file: Si_rt.data

units are all [a.u.]
c = light speed

1: t

2:$A_x^{ext}(t)$/c, 3:$A_y^{ext}(t)$/c, 4:$A_z^{ext}(t)$/c ← External Vector Potential

5:$E_x^{ext}(t)$ 6:$E_y^{ext}(t)$ 7:$E_z^{ext}(t)$ ← External Electric Field

8:$A_x^{tot}(t)$/c 9:$A_y^{tot}(t)$/c 10:$A_z^{tot}(t)$/c ← Total (External+Induced) Vector Potential

11:$E_x^{tot}(t)$ 12:$E_y^{tot}(t)$ 13:$E_z^{tot}(t)$ ← Total (External+Induced) Electric Field

14:$J_x^m(t)$ 15:$J_y^m(t)$ 16:$J_z^m(t)$ ← Current Density of Matter

17:$E_{tot}(t)$ 18:$E_{tot}(t) - E_{tot}(0)$ 19:$T_{ion}(t)$

Total Energy Excitation Energy Kinetic Energy of Ion

```
%> gnuplot
```

```
gnuplot> p “Si_rt.data” u 1:($18*27.2114) w l
```
→ Excitation Energy as a function of time

```
gnuplot> p “Si_rt.data” u 1:7 w l
```
→ Incident Electric Field (z-component)

```
gnuplot> p “Si_rt.data” u 1:16 w l
```
→ Current Density as a function of t (z-component)