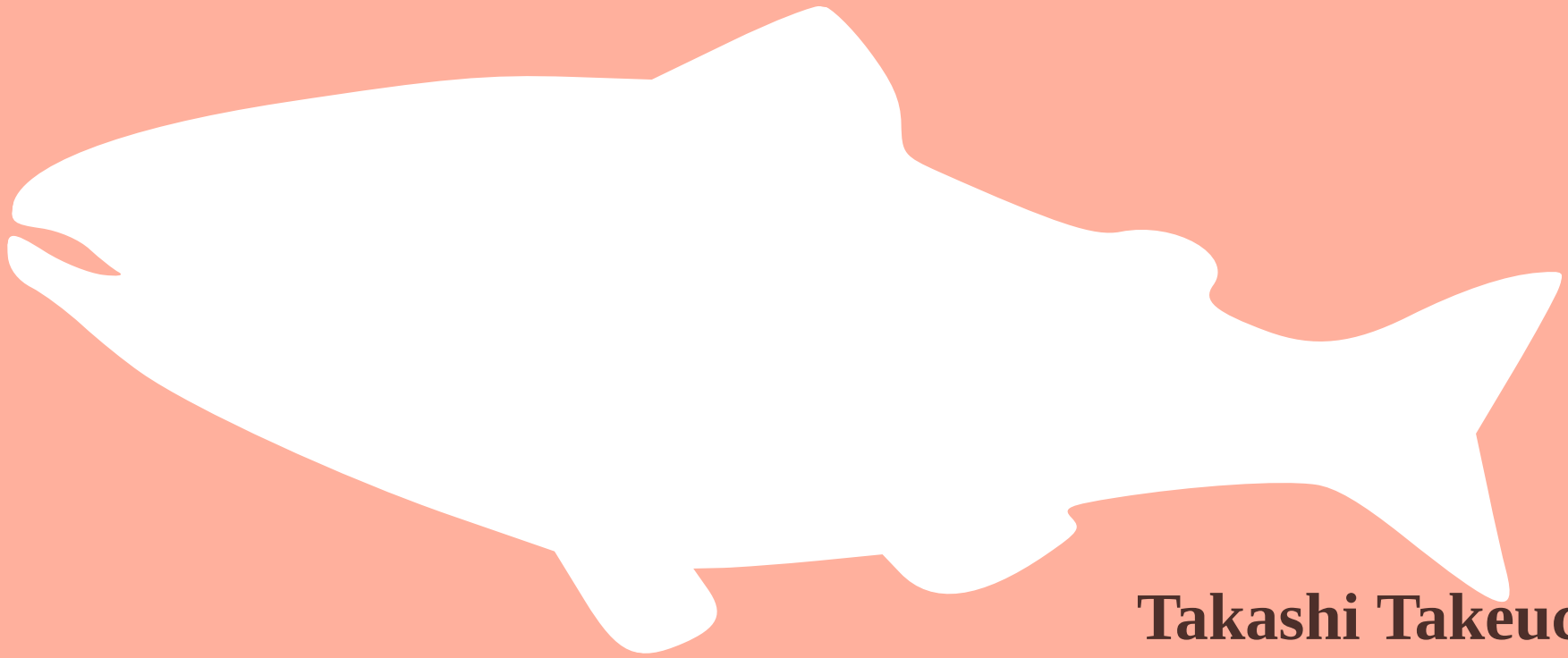
How to Use SALMON-1: Isolated Systems

Exercise



Takashi Takeuchi
Institute for Molecular Science

Demonstration for C_2H_2

- **Linear response calculation**:
  - Ground state calculation → C2H2_gs.inp
  - Real-time calculation → C2H2_rt_response.inp

- **Pulse response calculation**:
  - Ground state calculation → C2H2_gs.inp
  - Real-time calculation → C2H2_rt_pulse.inp

Demonstration for C_2H_2

- **Linear response calculation**:
  - **Ground state calculation → C2H2_gs.inp**
  - Real-time calculation → C2H2_rt_response.inp

- **Pulse response calculation**:
  - Ground state calculation → C2H2_gs.inp
  - Real-time calculation → C2H2_rt_pulse.inp

Submit job

- `cp -r /work/SALMON/tutor02/tutorial/1_isolated/ .`
- `cd 1_isolated/1_C2H2_gs_rt_response/`
- `cp /work/SALMON/tutor02/tutorial/job.sh ./job_gs.sh`
- `cp /work/SALMON/tutor02/tutorial/job.sh ./job_rt.sh`
- `sbatch job_gs.sh`
- `squeue -u <UserName>`

```
#!/bin/bash
#SBATCH -J gs
#SBATCH -p salmon
#SBATCH -N 4
#SBATCH --ntasks-per-node=2
#SBATCH --cpus-per-task=10
#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=./C2H2_gs.inp
program=salmon.cpu
mpirun -np $SLURM_NTASKS $program < $input > out_gs.log
```

Ground state calculation by C2H2_gs.inp

- &units /
- &calculation /
- &control /
- &system /
- &pseudo /
- &rgrid /
- &scf /
- &atomic_coor /

Basic rules for an input file

```
&category
keyword = xxx
/
```

→ The input file consists of category and keyword with an input value and equal sign(=).
→ A character input value requires single quotation marks (‘ ’).
→ A real input value can be written in Fortran style.
→ e.g. `1.0d-5`
→ “ ! ” is used as comment out.
→ Space free.
→ “ &atomic_coor/atomic_red_coor ” has to be set as the last category.
→ All keywords are described in the SALMON manual.

Ground state calculation by C2H2_gs.inp

- **&units /** 
- &calculation /
- &control /
- &system /
- &pseudo /
- &rgrid /
- &scf /
- &atomic_coor /

unit_system = 'A_eV_fs'
→ Unit system by Å, eV, and fs.
→ Default is atomic unit system.

Ground state calculation by C2H2_gs.inp

- &units /
- **&calculation /** ⇨
- &control /
- &system /
- &pseudo /
- &rgrid /
- &scf /
- &atomic_coor /

calc_mode = 'GS'
→ Ground state calculation.

Ground state calculation by C2H2_gs.inp

- &units /
- &calculation /
- **&control /** ⇒
- &system /
- &pseudo /
- &rgrid /
- &scf /
- &atomic_coor /

sysname = 'C2H2'
→ Name of calculation.
→ This is used for a prefix of output files.

Ground state calculation by C2H2_gs.inp

- &units /
- &calculation /
- &control /
- **&system /** ⟹
- &pseudo /
- &rgrid /
- &scf /
- &atomic_coor /

iperiodic = 0
→ Dimension for periodic boundary condition.
→ 0 is for isolated systems.

al = 16d0, 16d0, 16d0
→ Computational domain length.

nstate = 5
→ Number of states.

nelem = 2
→ Number of elements that will be used in calculations.

natom = 4
→ Number of atoms.

nelec = 10
→ Number of valence electrons.

Ground state calculation by C2H2_gs.inp

- &units /
- &calculation /
- &control /
- &system /
- **&pseudo /** ⟹
- &rgrid /
- &scf /
- &atomic_coor /

izatom(1) = 6, izatom(2) = 1
→ Atomic number.

pseudo_file(1) = 'C.cpi', pseudo_file(2) = 'H.cpi'
→ Name of pseudopotential files.

lmax_ps(1) = 1, lmax_ps(2) = 0
→ Maximum angular momentum of pseudopotential projectors.

lloc_ps (1) = 1, lloc_ps(2) = 0
→ Angular momentum of pseudopotential that will be treated as local.

- $V_{ion} = V_{local} + \sum_{lm} \frac{|\psi_{lm}^{PS} v_l\rangle\langle v_l \psi_{lm}^{PS}|}{\langle \psi_{lm}^{PS}|v_l|\psi_{lm}^{PS}\rangle}$

V_{local}: local part of pseudopotential

v_l: non-local part of pseudopotential

ψ_{lm}^{PS}: pseudowavefunction

Ground state calculation by C2H2_gs.inp

- &units /
- &calculation /
- &control /
- &system /
- &pseudo /
- **&rgrid /** 
- &scf /
- &atomic_coor /

dl = 0.25d0, 0.25d0, 0.25d0
→ Spacing of real-space grids.

Ground state calculation by C2H2_gs.inp

- &units /
- &calculation /
- &control /
- &system /
- &pseudo /
- &rgrid /
- **&scf /** 
- &atomic_coor /

ncg = 4
→ Number of iteration of Conjugate-Gradient method for each scf-cycle.
→ Default is 5.

nscf = 1000
→ Number of maximum scf cycle.

convergence = 'norm_rho_dng'
→ Convergence is checked by $||\text{rho\_iter(ix)-rho\_iter}_{-1}\text{(ix)}||^2$ / (number of grids).
→ Default is rho_dne
$\text{sum\_ix}|\text{rho(ix,iter)-rho(ix,iter-1)}|^2\text{dx}$.

threshold_norm_rho = 1.d-15
→ Threshold for convergence check.
→ Default is 1d-17.

Ground state calculation by C2H2_gs.inp

- &units /
- &calculation /
- &control /
- &system /
- &pseudo /
- &rgrid /
- &scf /
- **&atomic_coor /** ⇨

	x ↓	y ↓	z ↓	
'C'	0.000000	0.000000	0.599672	1
'H'	0.000000	0.000000	1.662257	2
'C'	0.000000	0.000000	-0.599672	1
'H'	0.000000	0.000000	-1.662257	2

→ **Positions of atoms written in Cartesian coordinates.**

Result of GS calculation by C2H2_gs.inp

- variables.log
  → Information of input values.
- out_gs.log
  → Standard output file.
- C2H2_info.data
  → System information including number of states, electrons, and energy, etc..
- C2H2_gs.bin
  → Temporary file used for RT calculation.
- C2H2_eigen.data
  → 1-particle energies

Result of GS calculation by C2H2_gs.inp

- $ cat out_gs .log

```
------------------------------------------------------------
iter =    43      Total Energy =      -340.06681205     Vh iteration =   1

     1       -18.4623       2       -14.0001       3       -12.3943       4       -7.3428

     5        -7.3428

iter and ||rho_i(ix)-rho_i-1(ix)||**2/(# of grids) =     43 0.52144155E-15

 Ne=    10.0000000000000
```

iteration count

Iteration count for V_h

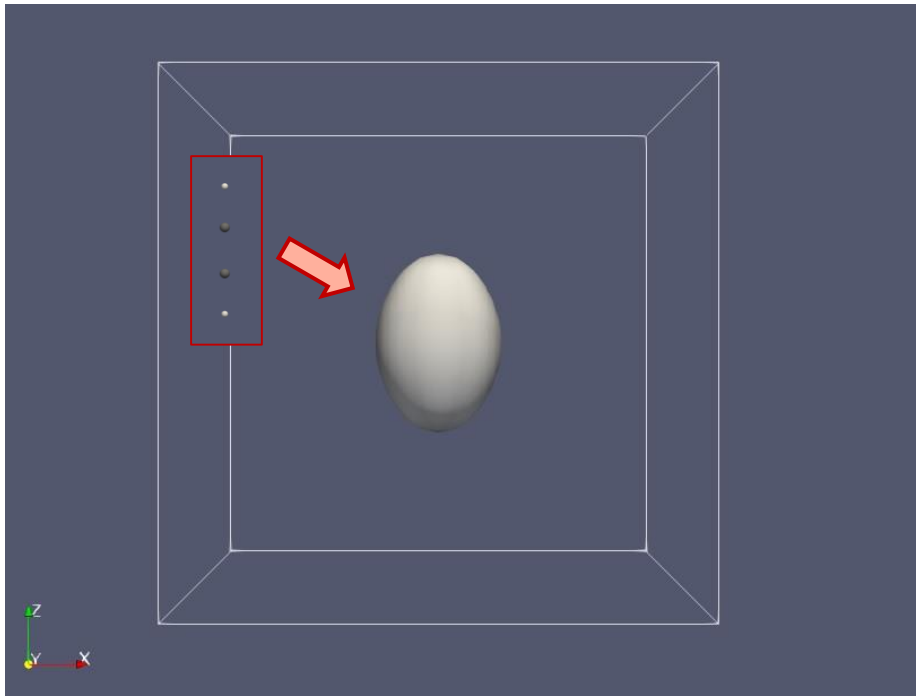
1-particle energies

Number of electros

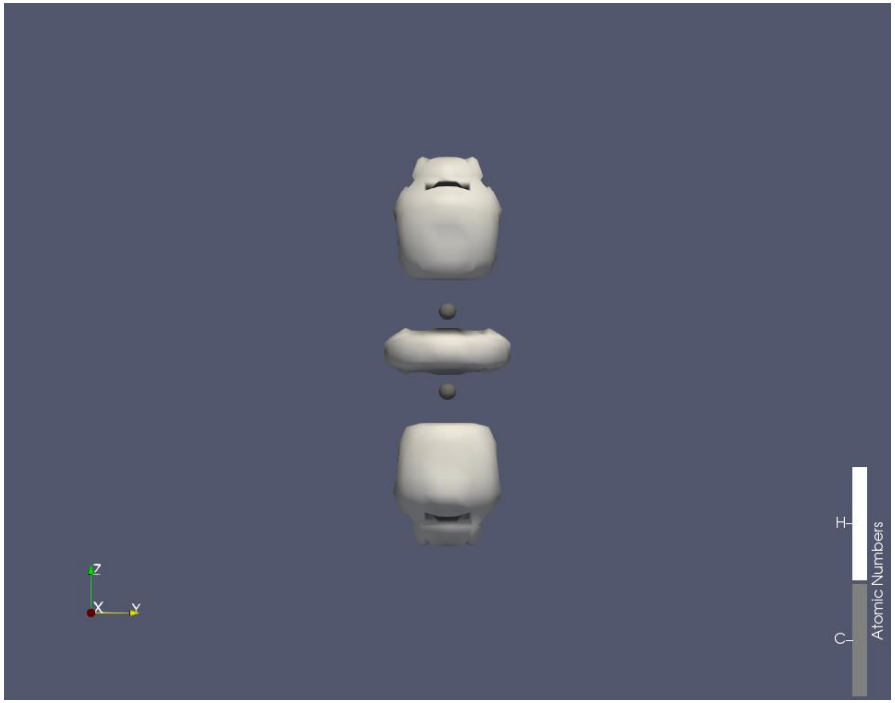
Convergence check

Result of GS calculation by C2H2_gs.inp

Electron density



Electron localized function



Demonstration for C_2H_2

- **Linear response calculation**:
  - Ground state calculation → C2H2_gs.inp
  - **Real-time calculation → C2H2_rt_response.inp**

- **Pulse response calculation**:
  - Ground state calculation → C2H2_gs.inp
  - Real-time calculation → C2H2_rt_pulse.inp

Submit job

- `sbatch job_rt.sh`
- `squeue -u <UserName>`

Real-time calculation by C2H2_rt_response.inp

- &units /
- &calculation /
- &control /
- &system /
- &pseudo /
- &tgrid /
- &emfield /
- &atomic_coor /

Real-time calculation by C2H2_rt_response.inp

- &units /
- **&calculation /** ⇨
- &control /
- &system /
- &pseudo /
- &tgrid /
- &emfield /
- &atomic_coor /

calc_mode = 'RT'
→ Real time calculation.

Real-time calculation by C2H2_rt_response.inp

- &units /
- &calculation /
- &control /
- &system /
- &pseudo /
- **&tgrid /** 
- &emfield /
- &atomic_coor /

dt = 1.25d-3
→ Time step.

nt = 5000
→ Number of total time steps for real-time propagation.

Real-time calculation by C2H2_rt_response.inp

- &units /
- &calculation /
- &control /
- &system /
- &pseudo /
- &tgrid /
- **&emfield /** 
- &atomic_coor /

ae_shape1 = 'impulse'
→ Impulsive fields.

epdir_re1 = 0.d0,0.d0,1.d0
→ Real part of polarization vector the pulse.

Result of GS calculation by C2H2_gs.inp

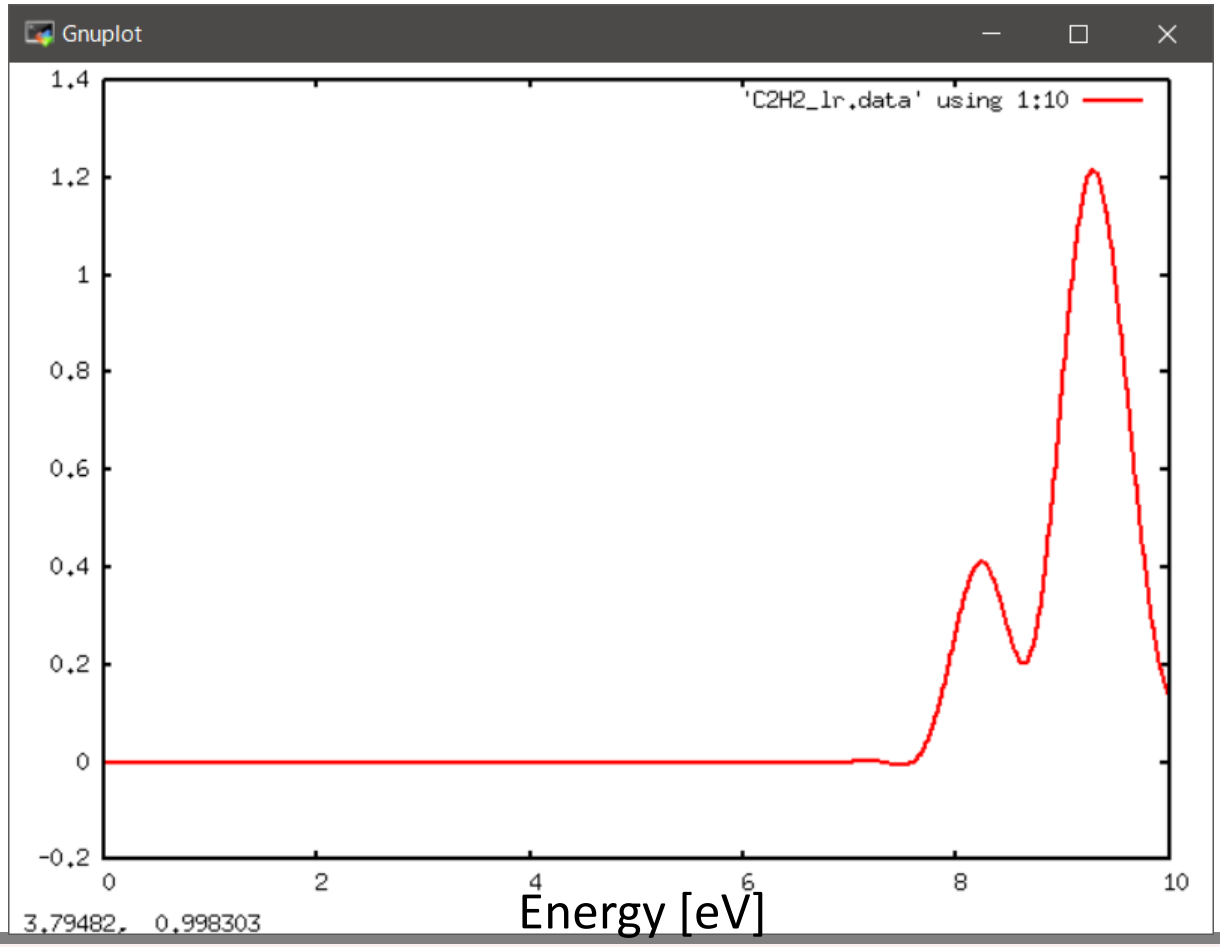
- out_rt.log
  → Standard output file.

```
timestep   time[fs]    Dipole moment(xyz)[A]      electrons         Total energy[eV]    iterVh

 ----------------------------------------------------

       1    0.00125000  0.29011063E-09  0.17076226E-09  0.27413687E-02    10.00000000     -340.05311442   97
```

- C2H2_p.data
  →The time response for the dipole moment.
- C2H2_lr.data
  → The dipole moment in frequency domain and Oscillator strength.

Result of RT calculation by C2H2_rt_response.inp

- `$ gnuplot`
- `$ plot 'C2H2_lr.data' using 1:10 w l lw 2`



Demonstration for C_2H_2

- **Linear response calculation**:
  - Ground state calculation → C2H2_gs.inp
  - Real-time calculation → C2H2_rt_response.inp

- **Pulse response calculation**:
  - Ground state calculation → C2H2_gs.inp
  - **Real-time calculation → C2H2_rt_pulse.inp**

Submit job

- `cd ../2_C2H2_gs_rt_pulse/`
- `cp /work/SALMON/tutor02/tutorial/job.sh ./job_gs.sh`
- `cp /work/SALMON/tutor02/tutorial/job.sh ./job_rt.sh`

- `sbatch job_gs.sh`
- `sbatch job_rt.sh`
- `squeue -u <UserName>`

Real-time calculation by C2H2_rt_pulse.inp

- &units /
- &calculation /
- &control /
- &system /
- &pseudo /
- &tgrid /
- &emfield /
- &atomic_coor /

Real-time calculation by C2H2_rt_pulse.inp

- &units /
- &calculation /
- &control /
- &system /
- &pseudo /
- **&tgrid /** 
- &emfield /
- &atomic_coor /

dt = 1.25d-3
→ Time step.

nt = 4800
→ Number of total time steps for real-time propagation.

Real-time calculation by C2H2_rt_pulse.inp

- &units /
- &calculation /
- &control /
- &system /
- &pseudo /
- &tgrid /
- **&emfield /** ⇨
- &atomic_coor /

ae_shape1 = 'Ecos2'
→ Envelope of $\cos^2$ for a scalar potential.

epdir_re1 = 0.d0,0.d0,1.d0
→ Real part of polarization vector the pulse.

rlaser_int_wcm2_1 = 1.d8
→ Peak laser intensity (W/cm^2) the pulse.

omega1 = 9.28d0
→ Mean photon energy of the pulse.

pulse_tw1 = 6.d0
→ Duration of the pulse.

phi_cep1 = 0.75d0
→ Carrier envelope phase of the pulse.

Result of RT calculation by C2H2_rt_pulse.inp

C2H2_ps.data
→ The dipole moment in frequency domain and the power spectrum.

- `$ gnuplot`
- `$ plot 'C2H2_p.data' using 1:4 w l lw 2`