Shiftk Documentation

Release 1.0.0

kawamura

February 15, 2017

Contents

Overview

This is document for the sample program which uses K ω library in the ISSP Math Library; this program computes the Green's function with $K\omega$. For the details of K ω library, See "$K\omega$ manual" in this package.

1.1 Calculation in this program

This program compute the Green's function

$$G_i(z) = \langle i|(z - \hat{H})^{-1}|i\rangle \equiv \boldsymbol{\varphi}_i^* \cdot (z - \hat{H})^{-1}\boldsymbol{\varphi}_i, \tag{1.1}$$

where $|i\rangle$ is a wavefunction, $\mathcal{H}$ is the Hamiltonian, and z is a complex frequency.

$\mathcal{H}$ **in the above equation is obtained by either the** following two ways:

- Input $\mathcal{H}$ as a file with the MatrixMarket format

- Construct $\mathcal{H}$ as a Hamiltonian of the Heisenberg model in this program.

In the computation of the Green's function, we use either the following two method according to the type of $\hat{H}$ (a real- or a complex- number).

- $\hat{H}$ of real numbers : Shifted Bi-Conjugate Gradient(BiCG) method

- $\hat{H}$ of complex numbers : Shifted Conjugate Orthogonal Conjugate Gradient(COCG) method

License

This software is developed under the support of
"*Project for advancement of software usability in materials science*" by The
Institute for Solid State Physics, The University of Tokyo.

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.
This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

For more details, See ' COPYING.LESSER ' in the root directory of this library.

Prerequisite

- fortran compiler
- K ω library
- BLAS, LAPACK
- MPI library (Optional)

Overall procedure

This program performs simulation as follows (See Fig. 4.1):

1. Read a input file for the numerical condition.

2. (Optional) Read input files for the Hamiltonian and the right hand side vector.

3. (Optional) Read a file for the restart (α, β, residual vectors)

4. (Bi)CG iteration with $K\omega$ library

5. Output results (Green's function, residual vectors, α, β)

We introduce the detailed procedure as follows. The format of each file is written in *Format of files*.
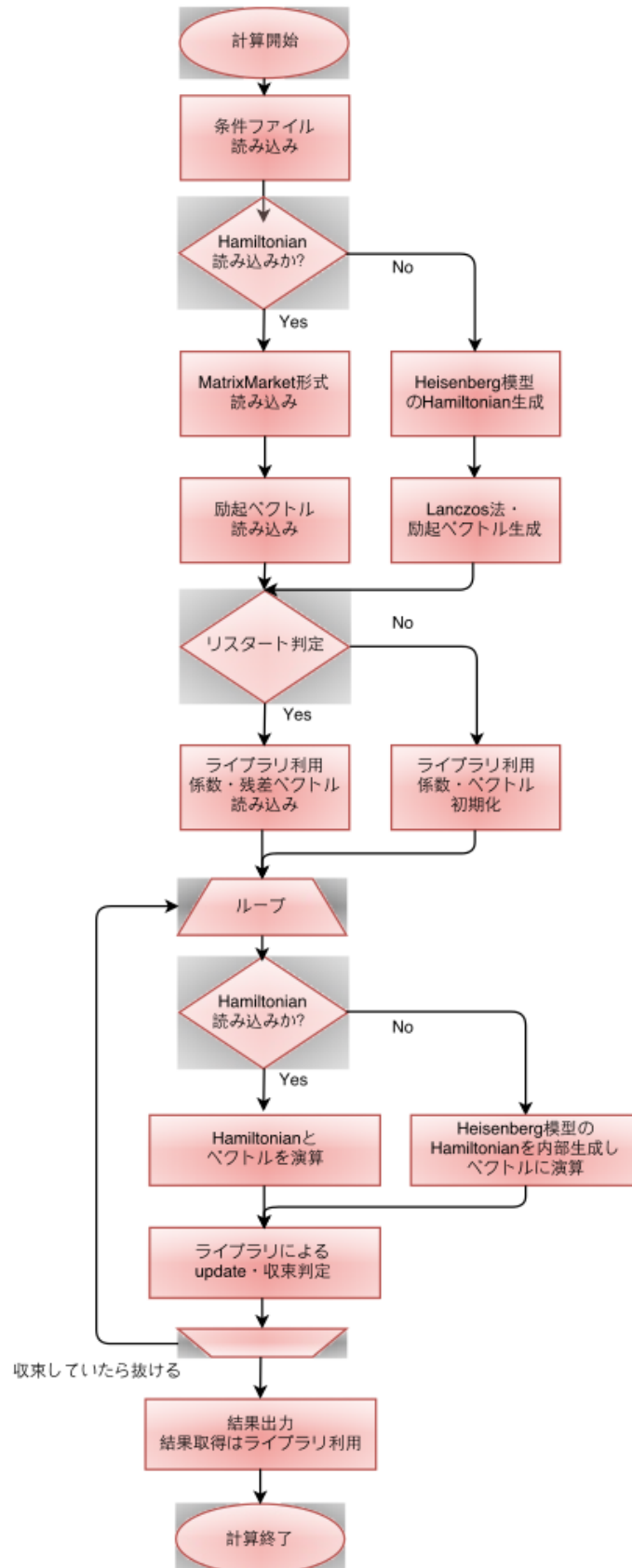
4.1 Calculation from scratch

4.1.1 Prepare input file

In this input file, we specify the frequency grid, the maximum number of iteration, etc.

A example of this input file is shown as follow.

```
&filename
  inham = ""
  invec = ""
/
&ham
nsite = 4
Jx = 1d0
Jy = 1d0
Jz = 1d0
Dz = 0d0
/
&cg
  maxloops = 100
  convfactor = 6
/
&dyn
  calctype = "normal"
  nomega = 100
  omegamin = (-2d0, 0.1d0)
  omegamax = ( 1d0, 0.1d0)
```

Figure 4.1: Schematic flow of the calculation of this program.

```
   outrestart = .TRUE.
/
```

where `InHam` is the file-name for the Hamiltonian in the MatrixMarket format, `InVec` is the file-name of the right hand side vector, `NOmega` is the number of frequencies where the dynamical Green's function is computed. `OmegaMin` is the lower limit of the frequency z, `OmegaMax` is the upper limit of the frequency z. These complex frequencies must be written as "`({Real part}, {Imaginaly part})`". The frequency grid is defined as

$$\omega_n = \texttt{OmegaMin} + \frac{n-1}{\texttt{NOmega}-1}(\texttt{OmegaMax} - \texttt{OmegaMin}). \tag{4.1}$$

`MaxLoops` is the maximum number of iterations. For the details of this input file, see *"Namelist" file*. The string variable must be specified with two `'` or `"`.

4.1.2 (Optional) Prepare input file for the Hamiltonian and RHS vector

Prepare input files for the Hamiltonian (MatrixMarket format) and right hand side vector.

An example of the file of the right hand side vector is shown as follows:

```
256
0.01    0.0
0.05    0.0
0.1     0.0
(continue ...)
```

Details of the file format is written in *"InHam" file* and *"InVec" file*.

4.1.3 Launch

Execute a program `ShiftK.out` with a namelist file `namelist.def` as follows:

```
$ ShiftK.out namelist.def
```

4.1.4 Output

The following files are generated in `output/` directory (this directory is also generated by this program).

```
TriDiagComp.dat, ResVec.dat, dynamicalG.dat
```

where `TriDiagComp.dat` contains the parameters for the restart, `ResVec.dat` contains the residual vector, `dynamicalG.dat` contains the dynamical Green's function. The format of these file is written in *Residual vector - Dynamical Green's function*.

4.2 Catch up

Compute the Green's function at the different frequency by using the information of `output/TriDiagComp.dat` that are generated in the previous run. This calculation does not require the matrix-vector product.

4.2.1 Modify the namelist file

Change `calctype` parameter in the file `namelist.def` from `"normal"` to `"recalc"`. `OmegaMin`, `OmegaMax`, etc. can be modified.

4.2.2 Launch

Execute a program `ShiftK.out` with a namelist file `namelist.def` as follows:

```
$ ShiftK.out namelist.def
```

The `output/dynamicalG.dat` will be overwritten.

4.2.3 Output

The same output files as the previous section are generated.

4.3 Restart

Compute the Green's function at the different frequency by using the information of `output/TriDiagComp.dat` and `output/ResVec.dat` that are generated in the previous run.

4.3.1 Modify the namelist file

Change `calctype` parameter in the file `namelist.def` from `"normal"` to `"restart"`. `OmegaMin`, `OmegaMax`, etc. can be modified. Parameters `InHam` , `InVec`, and the files of the Hamiltonian and RHS vector can not be modified.

4.3.2 Launch

Execute a program `ShiftK.out` with a namelist file `namelist.def` as follows:

```
$ ShiftK.out namelist.def
```

The `output/dynamicalG.dat` will be overwritten.

4.3.3 Output

The following files are generated (or are overwritten) in `output/` directory.

`TriDiagComp.dat, ResVec.dat, dynamicalG.dat`

where `TriDiagComp.dat` contains the parameters for the restart, `ResVec.dat` contains the residual vector, `dynamicalG.dat` contains the dynamical Green's function. The format of these file is written in *Residual vector - Dynamical Green's function*.

Format of files

5.1 Input files

5.1.1 "Namelist" file

This file has four namelist (`filename`, `ham`, `cg`, `dyn`). For the details of the namelist of fortran, see some books and Web page for fortran.

```
&filename
  inham = ""
  invec = ""
/
&ham
nsite = 4
Jx = 1d0
Jy = 1d0
Jz = 1d0
Dz = 0d0
/
&cg
  maxloops = 100
  convfactor = 6
/
&dyn
  calctype = "normal"
  nomega = 100
  omegamin = (-2d0, 0.1d0)
  omegamax = ( 1d0, 0.1d0)
  outrestart = .TRUE.
/
```

In the namelist *"filename"'*, we specify the file names of the Hamiltonian and the RHS vector.

- `InHam`

  Type : string (default: `""`)

  Description : The name of the file for the Hamiltonian in the MatrixMarket format (full path or relative path). If this parameter is not specified, the Hamiltonian of the 1D spin chain is generated with the parameters in `ham` namelist.

- `InVec`

  Type : string (default: `""`)

Description : The name of the file for the right hand side vector. If this parameter is not specified, the RHS vector is generated as follows: First the ground-state vector is computed. Then RHS vector is calculated by operating $\hat{S}_{1z}$ (z component of the spin at site 1) to the ground-state vector. Therefore, we assume the computed system is the one-dimensional spin chain.

"ham" namelist is read only when InHam is not specified. The parameters in this namelist are used to generate the Hamiltonian of the one-dimensional spin chain (See Fig. 5.1)

$$\hat{H} = \sum_i \begin{pmatrix} \hat{S}_{ix} & \hat{S}_{iy} & \hat{S}_{iz} \end{pmatrix} \begin{pmatrix} J_x & D_z & 0 \\ -D_z & J_y & 0 \\ 0 & 0 & J_z \end{pmatrix} \begin{pmatrix} \hat{S}_{i+1x} & \hat{S}_{i+1y} & \hat{S}_{i+1z} \end{pmatrix}. \tag{5.1}$$
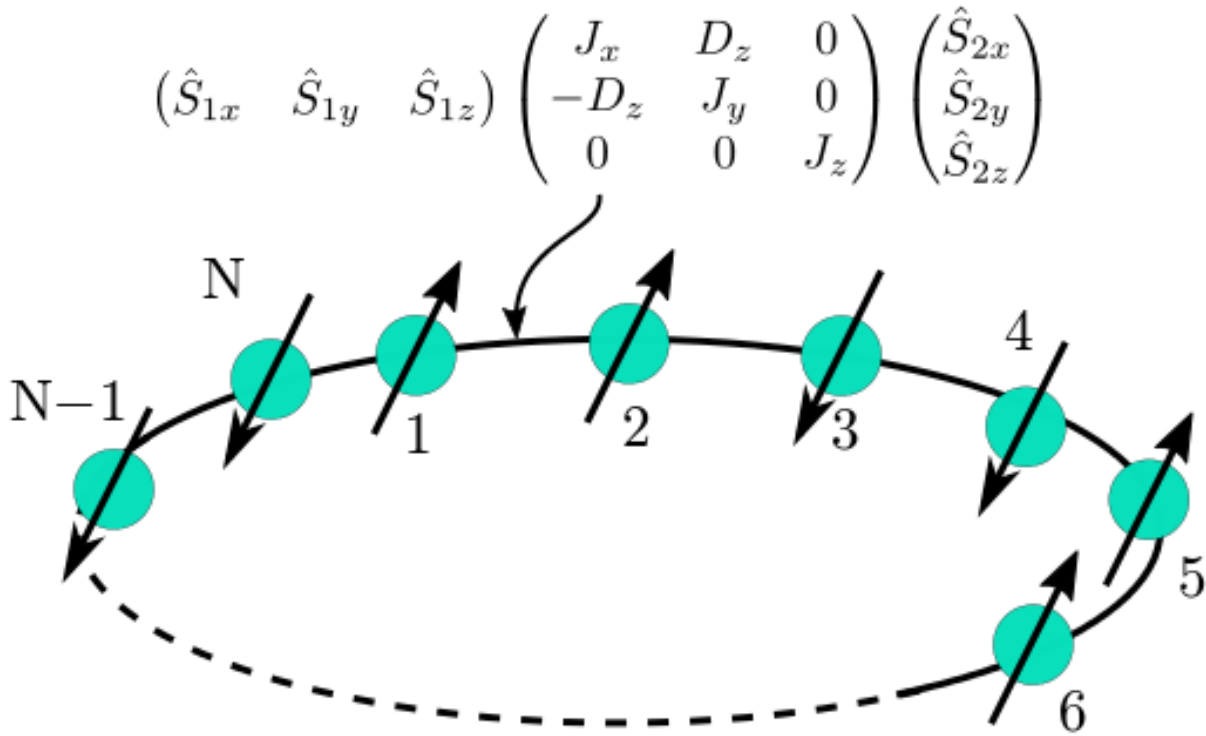

Figure 5.1: Schematic picture of the one-dimensional spin chain.

- nsite

  Type : int (default: 4)

  Description : The number of sites in the 1D spin chain model.

- Jx

  Type : double (default: 1.0)

  Description : J_x for the Heisenberg model.

- Jy

  Type : double (default: 1.0)

  Description : J_y for the Heisenberg model.

- Jz

  Type : double (default: `1.0`)

  Description : J_z for the Heisenberg model.

- Dz

  Type : double (default: `0.0`)

  Description : The parameter of the Dzyaloshinskii-Moriya interaction (D_z).

`cg` namelist has parameters for specifying the numerical condition for the (Bi)Conjugate-gradient method.

- MaxLoops

  Type : int (default: the dimension of the Hamiltonian)

  Description : The maximum number of iterations.

- Convfactor

  Type : int (default: `8`)

  Description : Convergence threshold. If the 2-norm of the residual vector becomes smaller than $10^{-\texttt{Convfactor}}$, the calculation finishes.

`dyn` has the parameters for the computation of the spectrum.

- OmegaMin

  Type : Complex (default: If `invec` is not specified, the real part of it becomes the smallest eigenvalue, and the imaginary part of it becomes $\times 0.01$ times the difference between the largest- and the smallest- eigenvalue. If `invec` is specified, the default value becomes `(0.0, 0.01)`)

  Description : The smallest frequency.

- OmegaMax

  Type : Complex (default: If `invec` is not specified, the real part of it becomes the largest eigenvalue, and the imaginary part of it becomes $\times 0.01$ times the difference between the largest- and the smallest- eigenvalue. If `invec` is specified, the default value becomes `(1.0, 0.01)`)

  Description : The largest frequency.

- NOmega

  Type : int (default: `10`)

  Description : The number of frequencies.

- outrestart

  Type : Logical (default: `.FALSE.`)

  Description : Whether data for restart will be written (`.TRUE.`) or not (`.FALSE.`).

- calctype

  Type : String. Choose from `"normal"`, `"recalc"`, `"restart"`. (default: `"normal"`)

  Description : `"normal"` for computing with the Krylov subspace from scratch. `"recalc"` for computing with the Krylov subspace generated in the previous calculation. In this case, the matrix-vector product is not performed. The convergence is not guaranteed. `"restart"` for restarting the calculation from the previous run. In this case, the calculation continues until the convergence is achieved or the number of iteration becomes `MaxLoops`.

5.1.2 "InHam" file

The Hamiltonian is written in the MatrixMarket format.

Example of the "InHam" file:

```
%%MatrixMarket matrix coordinate complex hermitian
16 16 20
1 1 1.000000 0.000000
3 2 0.500000 0.000000
5 3 0.500000 0.000000
6 4 0.500000 0.000000
6 6 -1.000000 0.000000
7 6 0.500000 0.000000
9 2 0.500000 0.000000
9 5 0.500000 0.000000
10 6 0.500000 0.000000
11 4 0.500000 0.000000
11 7 0.500000 0.000000
11 10 0.500000 0.000000
11 11 -1.000000 0.000000
12 8 0.500000 0.000000
13 6 0.500000 0.000000
13 11 0.500000 0.000000
14 12 0.500000 0.000000
15 8 0.500000 0.000000
15 14 0.500000 0.000000
16 16 1.000000 0.000000
```

First row: Arbitrary string (such as comment).

Second row: The number of rows, columns, and non-zero elements (in **the lower triangle part**).

Third row - End: The row- and the column- index, and the real- and imaginary- part of the non-zero element (in **the lower triangle part**).

5.1.3 "InVec" file

Right hand side vector is contained.

Example of the "InVec" file:

```
8192
0.02 0.01
0.02 0.001
(continue...)
```

- First row: [int01]

  Type : int

  Description : The dimension of the Hilbert space. It must be the same as the dimension of the Hamiltonian.

- Second row - End: [double01] [double02]

  Type : double

  Description : The element of the RHS vector. [double01] is the real part, and [double02] is the imaginary part.

5.1.4 Coefficients for restart

The coefficient α and β are contained.

The name of the generated file is `TriDiagComp.dat`.

The example of this file:

```
1000
1.0 0.0
0.1 0 0.01  0
0.2 0 0.021 0
(continue...)
2.1 -0.5
3.1 4.0
(continue...)
```

- First row: [int01]

  Type : int

  Description : The length of α, β. It is the same as the number of iterations in the previous run.

- Second row: [double01] [double02]

  Type : double

  Description : The seed frequency z_{seed}. [double01] is the real part of z_{seed}, and [double02] is the imaginary par of z_{seed}.

- Third row - 2 + [int01]th row: [double03] [double04] [double05] [double06]

  Type : double

  Description : α, β at the each iteration. [double03] is the real part of α, [double04] is the imaginary part of α, [double05] is the real part of β, [double06] is the imaginary part of β.

- 3 + [int01]th row - 2 + 2 $\times$ [int01]th row: [double07] [double08]

  Type : double

  Description : The product of the RHS vector and the residual vector at the each iteration. [double07] is the real part of it, and [double08] is the imaginary part of it.

5.1.5 Residual vector

The residual vector is contained for the restart. The file name is `ResVec.dat`.

Example of this file:

```
8192
0.02 0.01
0.02 0.001
(continue...)
0.02 0.01
0.02 0.001
(continue... Only for BiCG)
```

- First row: [int01]

  Type : int

  Description : The dimension of the Hilbert space.

- Second row - 1 + [int01]th row: [double01] [double02]

  Type : double

  Description : Each element of the residual vector. [double01] is the real part of it, and [double02] is the imaginary part of it.

- Third row - 1 + 2 × [int01]th row: [double03] [double04]

  Type : double

  Description : (Only when the Hamiltonian is a complex matrix) Each element of the shadow residual vector. [double03] is the real part of it, [double04] is the imaginary part of it.

5.2 Output file

5.2.1 Coefficient for restart

The format is the same as *Coefficients for restart*.

5.2.2 Residual vector

The format is the same as *Residual vector*.

5.2.3 Dynamical Green's function

The dynamical Green's function is contained.

Example of this file:

```
-10 0.001 0.001 -0.0001
-9.8 0.001 0.0012 -0.0002
-9.6 0.001 0.0014 -0.0003
(continue...)
```

- First row - END: [double01] [double02] [double03] [double04]

  - [double01], [double02]

    Type : double

    Description : The real- ([double01]) and the imaginary- ([double02]) part of the frequency.

  - [double03], [double04]

    Type : double

    Description : The dynamical Green's function. [double01] is the real part of it, and [double02] is the imaginary part of it.

Contact

If you have any comments, questions, bug reports etc. about this program, please contact to the main developer (Mitsuaki Kawamura) by sending the e-mail (the address is shown below).

```
mkawamura_at_issp.u-tokyo.ac.jp
```

Please change _at_ into @, when you will send the e-mail.