Table of Contents

1 About QS3

1.1 What is QS3?

1.2 License

1.3 Copyright

1.4 Developers

1.5 System Requirements

2 How to use QS3

2.1 Requirements

2.2 How to Install

2.3 Directory Structure

3 File Format

3.1 Required parameters

3.2 Parameters for lattices

3.3 Parameters for physical quantities

3.3.1 Local magnetization

3.3.2 Two-point correlation function

3.3.3 Dynamical spin structure factor

4 Algorithms

4.1 Lanczos Method

4.2 Thick-restarted Lanczos method

4.3 Continuous fraction expansion method

4.4 Calculation of physical quantities using wavenumber decomposition

Chapter 1

About QS3

1.1 What is QS3?

The exact diagonalization method is known as the most reliable method to evaluate the eigenstates and eigenvalues of a finite system without any approximation to the Hamiltonian. In the field of spin systems, various packages such as TITPACK, SPINPACK, and HPhi have been developed and widely used. However, the computational cost of the exact diagonalization method increases exponentially with the system size, so even if a system with S=1/2 is handled on a modern supercomputer, the accessible system size is limited up to 40 or so sites at most.

QS3 is an exact diagonalization package designed to study the low-energy states of the S=1/2 XXZ model, particularly near high magnetic fields. By using the translational and U(1) symmetries of the system, the computational cost is greatly reduced, and since it does not use bit operations, it can handle finite systems with more than a few hundred sites near saturated fields. QS3 implements the Thick-Restarted Lanczos method, which is a powerful method for false degeneracy problems, and can calculate basic physical quantities such as local magnetization and two-point correlation functions in a way that accurately captures low-energy excited states and their degenerate structures. QS3 also has a function to calculate the dynamical spin structure factor from the obtained ground state wavefunction using the continued fraction method. Although the conditions are very limited near the saturation field, it is possible to obtain exact spin structure factors with high resolution, and we hope that a wide range of users, including experimental researchers, will use it.

1.2 License

The complete source code for this software QS3 is distributed with MIT license. We would like users ask to cite the following reference when you have your paper with the results obtained from QS3.

H. Ueda, S. Yunoki, and T. Shimokawa, arXiv.XXXX.YYYY

1.3 Copyright

Copyright (c) 2021 QS-Cube

For the copyright of this software, please also see the following website.

<https://github.com/QS-Cube/ED/blob/main/LICENSE>

1.4 Developers

-ver. 1.0 (2021/XX/YY)

　 Hiroshi Ueda (Osaka University)

　Tokuro Shimokawa (Okinawa Institute of Science and Technology)

1.5 System Requirements

　・Linux PC / Mac + gfortran or intel fortran compiler

Chapter 2.

How to use QS3

* 1. Requirements

The QS3 software is written in Fortran 90, and uses BLAS and LAPACK as external libraries. The following items are required for compilation.

・gfortran compiler + BLAS/LAPACK library

・Intel fortran compiler + MKL library

2.2 How to install

If you have a Git account, please clone this repository on your local computer:

$ git clone <https://github.com/QS-Cube/ED.git>

If you don't have a Git account, go to the following site and click on "Code" buttom in the upper right corner of the site, then "Download ZIP" to download "ED-main.zip”.

<https://github.com/QS-Cube/ED>

Use the following command to extract the files.

$ unzip ED-main.zip

A Makefile to build the executable is provided in the source directory, and the user may need to change the path to BLAS/LAPACK or MKL appropriately. After changing the path, the following procedure will create the executable file and run some sample programs.

$ cd ED-main

$ cd script

$ ./make.sh

The calculation results will be output to the corresponding output folder.

Note that the following sample programs are available at the moment.

1. A program to calculate the low-energy states and various physical quantities in the total Sz=15, (kx, ky)=(0,0) sector of a 6×6 square lattice ferromagnet using the Thick-restarted Lanczos method under U(1) and translational symmetry.
2. A program that performs the calculation in 1) using the full diagonalization method.
3. A program to calculate the low-energy states and various physical quantities in the total Sz=15, (kx, ky)=(0,0) sector of a 10×10×10 cubic lattice antiferromagnet using the Thick-restarted Lanczos method under U(1) and translational symmetry.
4. A program to calculate the low-energy states and various physical quantities in the total Sz=15 sector of the 6×6 square lattice ferromagnet using the Lanczos method under only U(1) symmetry.
5. A program to calculate the low-energy states and various physical quantities in the total Sz=15, (kx, ky)=(0,0) sector of a 6×6 triangular lattice antiferromagnet using the Lanczos method under U(1) and translational symmetries.

2.3 Directory structure

The folders and files you will get after unzipping ED-main.zip are as follows. “input\_exX" contains the input files necessary to run the sample programs 1)-5) described above. The details of each file are described in the next section.

|-- input\_ex1/

|-- input.dat

|-- list\_cf\_ss.dat

|-- list\_local\_mag.dat

|-- list\_site\_position\_36\_type1.dat

|-- list\_xxz\_term\_36.dat

|-- input\_ex2/

|-- input.dat

|-- input\_ex3/

|-- input.dat

|-- list\_cf\_ss.dat

|-- list\_local\_mag.dat

|-- list\_site\_position\_1000\_type1.dat

|-- list\_xxz\_term\_1000.dat

|-- input\_ex4/

|-- input.dat

|-- input\_ex5/

|-- input.dat

|-- list\_xxz\_term\_36.dat

|-- script/

|-- make.sh

|-- source/

|-- Makefile

|-- eigen\_solver.f90

|-- get\_expectation\_values.f90

|-- ham2vec\_v3.f90

|-- input\_param.f90

|-- lanczos.f90

|-- main.f90

|-- state\_lists.f90

|-- source\_DSF/

|-- main.f90

|-- source\_mk\_input\_list/

|-- mk\_input\_list.f90

|-- source\_only\_u1/

|-- Makefile

|-- eigen\_solver.f90

|-- get\_expectation\_values.f90

|-- ham2vec\_v3.f90

|-- input\_param.f90

|-- lanczos.f90

|-- main.f90

|-- state\_lists.f90