

利用の手引き

高並列汎用格子模型ソルバー $\mathcal{H}\Phi$
ver. 0.2

January 22, 2016

Contents

1	What is $\mathcal{H}\Phi$?	1
1.1	$\mathcal{H}\Phi$ とは？	1
1.1.1	ライセンス	1
1.1.2	コピーライト	2
1.1.3	開発貢献者	2
1.2	動作環境	2
2	How to use $\mathcal{H}\Phi$?	3
2.1	要件	3
2.2	インストール方法	3
2.2.1	HPhiconfig.sh を使う方法	4
2.2.2	cmake を使う場合	5
2.3	ディレクトリ構成	6
2.4	基本的な使い方	8
2.4.1	スタンダードモード	8
2.4.2	エキスパートモード	10
2.4.3	プロセス数の設定	11
3	チュートリアル	12
3.1	スタンダードモード	12
3.1.1	Heisenberg 模型	12
3.1.2	その他の系でのチュートリアル	18
3.2	エキスパートモード	19
3.2.1	入力ファイルリストファイル	19
3.2.2	基本パラメータの指定	19
3.2.3	Hamiltonian の指定	21
3.2.4	出力ファイルの指定	23
3.2.5	計算の実行	24

3.2.6	その他の系でのチュートリアル	24
3.3	エキスパート用入力ファイル作成モード	25
4	ファイル仕様	26
4.1	スタンダードモード用入力ファイル	26
4.1.1	計算の種類に関する必須パラメーター	27
4.1.2	格子に関するパラメーター	27
4.1.3	保存量に関するパラメーター	29
4.1.4	Hubbard 模型パラメーター	29
4.1.5	Kitaev-Heisenberg 模型パラメーター	30
4.1.6	近藤格子模型パラメーター	31
4.1.7	計算条件のパラメーター	31
4.2	エキスパートモード用入力ファイル	34
4.2.1	入力ファイル指定用ファイル	35
4.2.2	CalcMod ファイル	36
4.2.3	ModPara ファイル	39
4.2.4	LocSpin 指定ファイル	43
4.2.5	Trans 指定ファイル	45
4.2.6	InterAll 指定ファイル	47
4.2.7	CoulombIntra 指定ファイル	49
4.2.8	CoulombInter 指定ファイル	51
4.2.9	Hund 指定ファイル	53
4.2.10	PairHop 指定ファイル	55
4.2.11	Exchange 指定ファイル	57
4.2.12	Ising 指定ファイル	59
4.2.13	PairLift 指定ファイル	61
4.2.14	OneBodyG 指定ファイル	63
4.2.15	TwoBodyG 指定ファイル	65
4.3	出力ファイル	67
4.3.1	CHECK_Chemi.dat	67
4.3.2	CHECK_InterAll.dat	68
4.3.3	CHECK_CoulombIntra.dat	69
4.3.4	CHECK_Hund.dat	70
4.3.5	CHECK_INTER_U.dat	71
4.3.6	CHECK_Memory.dat	72
4.3.7	WarningOnTransfer.dat	73

4.3.8	TimeKeeper.dat	74
4.3.9	sz_TimeKeeper.dat	74
4.3.10	Time_CG_EigenVector.dat	75
4.3.11	energy.dat	76
4.3.12	Lanczos_Step.dat	77
4.3.13	Norm_rand.dat	78
4.3.14	SS_rand.dat	79
4.3.15	Eigenvalue.dat	81
4.3.16	phys.dat	82
4.3.17	cisajs.dat	84
4.3.18	cisajscktalt.dat	86
4.3.19	eigenvec.dat	88
4.4	エラーメッセージ一覧	90
5	アルゴリズム	91
5.1	Lanczos 法	91
5.1.1	手法概要	91
5.1.2	逆反復法	92
5.1.3	実際の実装について	93
5.2	完全対角化	94
5.2.1	手法概要	94
5.2.2	有限温度物理量の計算	94
5.3	熱的純粋量子状態による有限温度計算	94
5.3.1	実際の実装について	95
6	謝辞	96

1

What is $\mathcal{H}\Phi$?

1.1 $\mathcal{H}\Phi$ とは？

実験データと理論模型の解析結果との直接比較は、物質科学の研究プロセスの一つの核となっています。例えば、低エネルギー励起構造を反映する比熱の温度依存性や、帯磁率から見積もった有効スピン・モーメントおよびキュリー・ワイス温度等の実験データからその背後にある電子状態を理解するためには、理論模型の解析結果との比較が必要不可欠です。

理論模型の解析手法としては、厳密対角化法 [1] が近似を用いず定量的な計算ができる最も信頼できる手法です。この目的のためには、東京工業大学の西森秀稔教授が作られた固有値問題ソルバパッケージ TITPACK が長年広く利用されてきました。これまでは計算機資源の制約もあり、その利用は比較的小さな系への適用に限られてきました。しかし、計算機の進歩によって、並列化を利用してハバード模型 18 格子点、スピン 1/2 の格子スピン模型 36 格子点程度の計算が容易に実行可能となっています。また、量子統計力学の発展 [2–5] により、基底状態計算と同程度のコストでの有限温度計算が可能となり、比熱や帯磁率の実験との定量的比較が十分可能となってきています [6]。これらの計算が可能になった背景には、低バンド幅で多数の分散メモリ計算コアを持つ新しい計算機アーキテクチャがあり、その性能を最大限に生かしつつ、簡便に利用可能な新しいソフトウェアの登場が待たれていました。

このような背景に基づき、汎用量子模型ソルバーパッケージ $\mathcal{H}\Phi$ は、ランチョス法に基づく量子多体模型の基底状態および低励起状態に対する厳密対角化法と、熱的純粋量子状態 [5] を基礎とした有限温度計算を、簡便かつ柔軟なユーザー・インターフェイスとともに並列実装されたアプリケーションを目指して開発しました。シンプルなハバード模型やハイゼンベルグ模型に始まり、多軌道ハバード模型やジャロシンスキー-守谷相互作用やキタエフ項のように $SU(2)$ 対称性を破る相互作用を持つ量子スピン模型、さらに遍歴電子と局在スピンが結合した近藤格子模型まで、ユーザーの興味に応じて広汎な量子格子模型を解析することができます。基底状態および有限温度での内部エネルギーはもちろん、比熱や電荷・スピン構造因子を始めとする様々な物理量が計算可能となっています。実験研究者を含む幅広いユーザーに気楽にご利用いただければ幸いです。

1.1.1 ライセンス

本ソフトウェアのプログラムパッケージおよびソースコード一式は GNU General Public License version 3 (GPL v3) に準じて配布されています。

1.1.2 コピーライト

©2015- T. Misawa, K. Yoshimi, M. Kawamura, Y. Yamaji, S. Todo
and N. Kawashima. All rights reserved.

なお、本ソフトウェアは2015年度 東京大学物性研究所 ソフトウェア高度化プロジェクトの支援を受け開発されています。

1.1.3 開発貢献者

本ソフトウェアは以下の開発貢献者により開発されています。

- ver.0.2 (2015/12/28 リリース)
- ver.0.1 (2015/10/09 リリース)
 - 開発者
 - * 三澤 貴宏 (東京大学大学院 工学系研究科)
 - * 河村 光晶 (東京大学 物性研究所)
 - * 吉見 一慶 (東京大学 物性研究所)
 - アドバイザー
 - * 山地 洋平 (東京大学大学院 工学系研究科)
 - * 藤堂 眞治 (東京大学 理学系研究科)
 - プロジェクトコーディネーター
 - * 川島 直輝 (東京大学 物性研究所)

1.2 動作環境

以下の環境で動作することを確認しています。

- 東京大学物性研究所スーパーコンピューターシステム B 「sekirei」
- 同システム C 「maki」
- Linux PC + intel コンパイラ
- Linux PC + gcc
- Mac + gcc

2

How to use $\mathcal{H}\Phi$?

2.1 要件

$\mathcal{H}\Phi$ のコンパイル・使用には次のものがが必要です。

- C コンパイラ (インテル、富士通、GNU など)
- LAPACK ライブラリ (インテル MKL, 富士通 TCL, ATLAS など)
- MPI ライブラリ (MPI 並列を行わない場合は必要ありません)

Tips

例/ intel コンパイラーでの設定

intel コンパイラを使用する場合には、コンパイラに付属の設定用スクリプトを使用するのが簡単です。

64 ビット OS で bash を使っている場合には

```
source /opt/intel/bin/compilervars.sh intel64
```

または

```
source /opt/intel/bin/iccvars.sh intel64
```

```
source /opt/intel/mkl/bin/mklvars.sh
```

等を ~/.bashrc に記載してください。詳しくはお手持ちのコンパイラ、ライブラリのマニュアルをお読みください。

2.2 インストール方法

$\mathcal{H}\Phi$ は次の場所からダウンロードできます。

<https://github.com/QLMS/HPhi/releases>

ダウンロードしたファイルを次のように展開してください。

```
$ tar xzvf HPhi-xxx.tar.gz
```

$\mathcal{H}\Phi$ は次の 2 通りの方法でインストールできます。

2.2.1 HPhiconfig.sh を使う方法

展開したディレクトリのなかにある HPhiconfig.sh スクリプトを次のように実行してください。(物性研システム B "sekirei" の場合)

```
$ bash HPhiconfig.sh sekirei
```

これによりコンパイル環境設定ファイル `make.sys` が `src/` ディレクトリに作られます。HPhiconfig.sh の引数は次のものに対応しています。

- `sekirei` : 物性研究所システム B "sekirei"
- `maki` : 物性研究所システム C "maki"
- `intel` : intel コンパイラ + Linux PC
- `mpicc-intel` : intel コンパイラ + MPI ライブラリ (intelMPI 以外) + Linux PC
- `gcc` : GCC + Linux PC
- `gcc-mac` : GCC + Mac

`make.sys` の中身は次のようになっています (物性研システム B "sekirei" の場合)。

```
CC = mpiicc
LAPACK_FLAGS = -Dlapack -mkl=parallel
FLAGS = -qopenmp -O3 -xCORE-AVX2 -mcmodel=large -shared-intel -D MPI
MTFLAGS = -DDSFMT_MEXP=19937 $(FLAGS)
INCLUDE_DIR=./include
```

となります。それぞれのマクロ (変数) の説明は次のとおりです。

- `CC` : コンパイルコマンド (`icc`, `gcc`, `fccpx`)
- `LAPACK_FLAGS` : LAPACK のためのコンパイルオプション。 `-Dlapack` は必須です。
- `FLAGS` : その他のコンパイルオプション。 `-openmp`, `-fopenmp`, `-qopenmp` などの OpenMP 有効化のオプションは必須です。MPI 並列を行う場合は `-D MPI` をつけます。
- `MTFLAGS`, `INCLUDE_DIR` : メルセンヌツイスター (乱数発生ルーチン) および追加のインクルードディレクトリを指定します。これらを変更する必要はありません。

これでコンパイルのための準備が整います。その後

```
$ make HPhi
```


とすることで実行可能ファイル HPhi が src/内に生成されるので、このディレクトリにパスを通すか、パスの通っている場所にシンボリックリンクを作ってください。

Tips

実行ファイルにパスを通す時には、次のようにします。

```
$ export PATH=${PATH}:HPhiのディレクトリ/src/
```

この設定を常に残すには、例えばログインシェルが bash の場合には ~/.bashrc ファイルに上記のコマンドを記載します。

2.2.2 cmake を使う場合

Tips

sekirei で cmake を利用するには

```
source /home/issp/materiapps/tool/env.sh
```

maki では

```
source /global/app/materiapps/tool/env.sh
```

をあらかじめ実行する必要があります。

HPhi を展開したディレクトリのパスを \$PathTohphi 、ビルドディレクトリを \$HOME/build/hphi (任意の場所を指定可能) とした場合に、

```
cd $HOME/build/hphi
cmake -DCONFIG=gcc $PathTohphi
make
```

でコンパイルすることができます。コンパイル後、\$HOME/build/hphi 直下に src フォルダが作成され、実行ファイルである HPhi がそのフォルダ内に作成されます。MPI ライブラリがない場合には、MPI 非対応の実行ファイルが作成されます。

なお、上の例では gcc コンパイラを前提としたコンパイルになっていますが、

- sekirei : 物性研究所システム B "sekirei"
- fujitsu : 富士通コンパイラ (物性研究所システム C "maki")
- intel : intel コンパイラ + Linux PC
- gcc : GCC + Linux PC

のオプションが用意されています。以下、HPhi を展開したディレクトリでビルドする例を示します (intel コンパイラの場合)。

```
mkdir ./build
cd ./build
cmake -DCONFIG=intel ../
make
```

実行後、build フォルダ直下に src フォルダが作成され、HPhi が src フォルダ内に作成されます。なお、コンパイラを変更しコンパイルし直したい場合には、都度 build フォルダごと削除を行った上で、新規に上記作業を行うことをお勧めします。

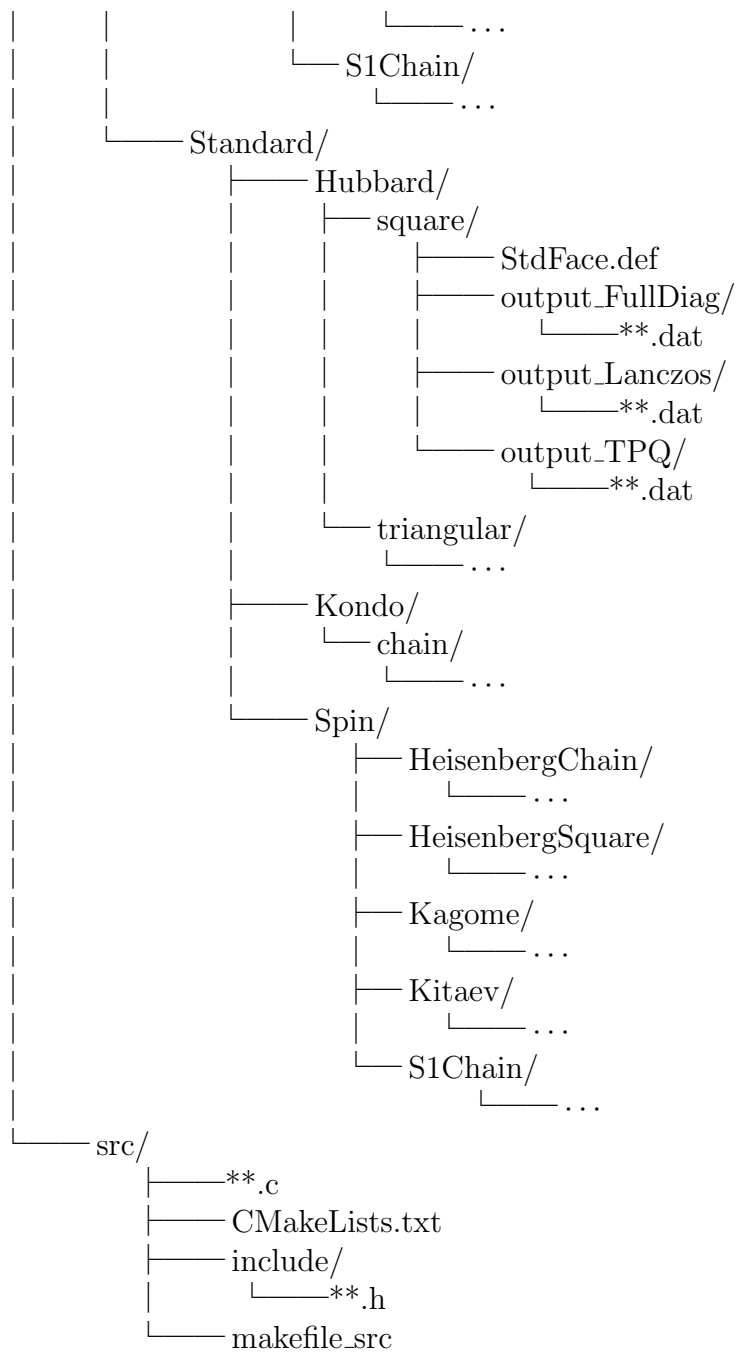
2.3 ディレクトリ構成

HPhi-xxx.gz を解凍後に構成されるディレクトリ構成を以下に示します。

```

|—— CMakeLists.txt
|—— COPYING
|—— config/
|   |—— fujitsu.cmake
|   |—— gcc.cmake
|   |—— intel.cmake
|   |—— sekirei.cmake
|—— doc/
|   |—— en/
|   |—— jp/
|   |—— userguide_en.pdf
|   |—— userguide_jp.pdf
|—— HPhiconfig.sh
|—— samples/
|   |—— Expert/
|   |   |—— Hubbard/
|   |   |   |—— square/
|   |   |   |   |—— *.def
|   |   |   |   |—— output_FullDiag/
|   |   |   |   |   |—— **.dat
|   |   |   |   |—— output_Lanczos/
|   |   |   |   |   |—— **.dat
|   |   |   |   |—— output_TPQ/
|   |   |   |   |   |—— **.dat
|   |   |   |—— triangular/
|   |   |   |   |—— ...
|   |   |—— Kondo/
|   |   |   |—— chain/
|   |   |   |   |—— ...
|   |   |—— Spin/
|   |   |   |—— HeisenbergChain/
|   |   |   |   |—— ...
|   |   |   |—— HeisenbergSquare/
|   |   |   |   |—— ...
|   |   |—— Kitaev/

```



2.4 基本的な使い方

$\mathcal{H}\Phi$ ではスタンダードモードとエキスパートモードの2つのモードが存在します。ここでは、スタンダードモードおよびエキスパートモードでの計算に関して、それぞれ基本的な流れを記載します。

2.4.1 スタンダードモード

スタンダードモードでの動作方法は下記の通りです。

1. 計算用ディレクトリの作成

計算シナリオ名を記載したディレクトリを作成します。

2. スタンダードモード用入力ファイルの作成

スタンダードモードでは、あらかじめ用意されたいくつかのモデル (Heisenberg モデルや Hubbard モデル) や格子 (正方格子など) を指定し、それらに対するいくつかのパラメーター (最近接・次近接スピン結合やオンサイトクーロン積分など) と計算手法 (Lanczos 法、TPQ 法など) を設定します。各ファイルは Sec. 4.1 に従い記載してください。

3. 実行

”-s”(“--standard” でも可) をオプションとして指定の上、1 で作成した入力ファイル名を引数とし、HPhi を実行します。

- シリアル/OpenMP 並列 の場合

```
$ パス/HPhi -s 入力ファイル
```

- MPI 並列/ハイブリッド並列 の場合

```
$ mpiexec -np プロセス数 パス/HPhi -s 入力ファイル
```

ワークステーションやスパコン等でキューイングシステムを利用している場合はプロセス数をジョブ投入コマンドの引数として与える場合があります。詳しくはお使いのシステムのマニュアルをご参照ください。プロセス数の指定に関しては計算する系により固定のものに設定する必要があります。詳細は 2.4.3 を参照ください。

4. 途中経過

計算実行の経過について output フォルダにログファイルが出力されます。出力されるファイルの詳細に関しては 4.3 を参考にしてください。

5. 最終結果

計算が正常終了した場合、計算モードに従い output フォルダに計算結果ファイルが出力されます。出力されるファイルの詳細に関しては 4.3 を参考にしてください。

Tips**OpenMP スレッド数の指定**

実行時の OpenMP のスレッド数を指定する場合は、 $\mathcal{H}\Phi$ を実行する前に以下の様にしてください (16 スレッドの場合)。

```
export OMP_NUM_THREADS=16
```

2.4.2 エキスパートモード

エキスパートモードでの動作方法は下記の通りです。

1. 計算用ディレクトリの作成

計算シナリオ名 (名前は任意) を記載したディレクトリを作成します。

2. 詳細入力ファイルの作成

エキスパートモードでは、ハミルトニアンすべての項を記述する詳細入力ファイルと計算条件のファイル、およびそれらのファイル名のリストファイルを作成します。各ファイルは Sec. 4.2 に従い記載してください。なお、リストファイルの作成はスタンダード用のファイル StdFace.def を用いると容易に作成することができます。

3. 実行

”-e”(“--expert” でも可) をオプションとして指定の上、1 で作成した入力リストファイル名を引数とし、ターミナルから $\mathcal{H}\Phi$ を実行します。

- シリアル/OpenMP 並列

```
$ パス/HPhi -e 入力リストファイル
```

- MPI 並列/ハイブリッド並列

```
$ mpiexec -np プロセス数 パス/HPhi -e 入力リストファイル
```

プロセス数の指定に関しては計算する系により固定のものに設定する必要があります。詳細は 2.4.3 を参照ください。

4. 途中経過

計算実行の経過について output フォルダにログファイルが出力されます。出力されるファイルの詳細に関しては 4.3 を参考にしてください。

5. 最終結果

計算が正常終了した場合、計算モードに従い output フォルダに計算結果ファイルが出力されます。出力されるファイルの詳細に関しては 4.3 を参考にしてください。

2.4.3 プロセス数の設定

MPI並列/ハイブリッド並列を用いる場合、プロセス数は以下のように設定してください。

1. Standard モード

- 電子系及び近藤格子系
スタンダードモード用入力ファイルで `model="Fermion Hubbard"`, `"Kondo Lattice"`, `"Fermion HubbardGC"` の場合は、プロセス数が 4^n となるように設定してください。
- スピン系
スタンダードモード用入力ファイルで `model="Spin"`, `"SpinGC"` の場合は、入力ファイルの `2S` の値に対してプロセス数が $(2S+1)^n$ となるように設定してください (デフォルトは `2S=1`)。

2. Expert モード

- 電子系及び近藤格子系 4.2.2 の **CalcMod** ファイルで、`CalcModel` として `fermion Hubbard` 模型、近藤模型を選択した場合は、プロセス数が 4^n となるように設定してください。
- スピン系
4.2.2 の **CalcMod** ファイルで、`CalcModel` としてスピン模型を選択した場合は、4.2.4 の **LocSpin** ファイルを参考にプロセス数を指定する必要があります。許容されるプロセス数は、サイト数の大きいものから順に局在スピンの状態数 $(2S+1)$ を掛けたもので指定されます。
例えば、**LocSpin** ファイルが

```
=====
NlocalSpin      3
=====
=====i_0IteElc_2S =====
=====
      0      3
      1      2
      2      1
```

で与えられる場合、許容されるプロセス数は $2 = 1+1$, $6 = 2 \times (2+1)$, $24 = 6 \times (3+1)$ となります。

3

チュートリアル

3.1 スタンダードモード

3.1.1 Heisenberg 模型

以下のチュートリアルはディレクトリ

`samples/Standard/Spin/HeisenbergChain/`

内で行います。Heisenberg 模型におけるサンプル入力ファイル、参照用出力ディレクトリ、標準出力リダイレクトはそれぞれ

```
samples/Standard/Spin/HeisenbergChain/StdFace.def
samples/Standard/Spin/HeisenbergChain/output_FullDiag/
samples/Standard/Spin/HeisenbergChain/output_Lanczos/
samples/Standard/Spin/HeisenbergChain/output_TPQ/
samples/Standard/Spin/HeisenbergChain/FullDiag.out
samples/Standard/Spin/HeisenbergChain/Lanczos.out
samples/Standard/Spin/HeisenbergChain/TPQ.out
```

にあります。この例では1次元の Heisenberg 鎖 (最近接サイト間の反強磁性的スピン結合のみを持つ) を考察します。

$$\hat{H} = J \sum_{i=1}^{N_{\text{site}}} \hat{\mathbf{S}}_i \cdot \hat{\mathbf{S}}_{i+1} \quad (3.1)$$

インプットファイルの中身は次のとおりです。

```
L = 16
model = "Spin"
method = "Lanczos"
lattice = "chain lattice"
J = 1.0
2Sz = 0
```

この例ではスピン結合 $J = 1$ (任意単位) とし、サイト数は16としました。

ログ出力

標準出力でログが出力されます。また、output ディレクトリが自動生成され、そこにも計算経過を示すログが出力されます。例えば、サンプル実行時には以下のファ

イルが出力されます。

CHECK_InterAll.dat	Time_CG_EigenVector.dat	zvo_Lanczos_Step.dat
CHECK_Memory.dat	WarningOnTransfer.dat	zvo_sz_TimeKeeper.dat
CHECK_Sdim.dat	zvo_TimeKeeper.dat	

ログ出力されるファイルの詳細は 4.3.1-4.3.10 をご覧ください。実行コマンドと標準出力 (MPI 並列/ハイブリッド並列でコンパイルした場合の結果) は次のとおりです。

\$ パス/HPhi -s StdFace.def

Parallelization Info.

OpenMP threads : 4
MPI PEs : 1

Standard Interface Mode STARTS

Open Standard-Mode Inputfile ./StdFace.def

KEYWORD : l	VALUE : 16
KEYWORD : model	VALUE : spin
KEYWORD : method	VALUE : lanczos
Skipping a line.	
Skipping a line.	
KEYWORD : lattice	VALUE : chainlattice
KEYWORD : j	VALUE : 1.0
KEYWORD : 2sz	VALUE : 0

Parameter Summary

L = 16		
a = 1.00000	#####	DEFAULT VALUE IS USED #####
2S = 1	#####	DEFAULT VALUE IS USED #####
h = 0.00000	#####	DEFAULT VALUE IS USED #####
Gamma = 0.00000	#####	DEFAULT VALUE IS USED #####
D = 0.00000	#####	DEFAULT VALUE IS USED #####
J = 1.00000		
Jz = 1.00000	#####	DEFAULT VALUE IS USED #####
Jxy = 1.00000	#####	DEFAULT VALUE IS USED #####
Jx = 1.00000	#####	DEFAULT VALUE IS USED #####
Jy = 1.00000	#####	DEFAULT VALUE IS USED #####
J' = 0.00000	#####	DEFAULT VALUE IS USED #####
Jz' = 0.00000	#####	DEFAULT VALUE IS USED #####

```

        Jxy' = 0.00000      ##### DEFAULT VALUE IS USED #####
        Jx'  = 0.00000      ##### DEFAULT VALUE IS USED #####
        Jy'  = 0.00000      ##### DEFAULT VALUE IS USED #####
    LargeValue = 1          ##### DEFAULT VALUE IS USED #####

        filehead = zvo      ##### DEFAULT VALUE IS USED #####
    Lanczos_max = 2000      ##### DEFAULT VALUE IS USED #####
        initial_iv = 1      ##### DEFAULT VALUE IS USED #####
            nvec = 1         ##### DEFAULT VALUE IS USED #####
            exct = 1         ##### DEFAULT VALUE IS USED #####
        LanczosEps = 14      ##### DEFAULT VALUE IS USED #####
    LanczosTarget = 2        ##### DEFAULT VALUE IS USED #####
        NumAve = 5          ##### DEFAULT VALUE IS USED #####
    ExpecInterval = 20       ##### DEFAULT VALUE IS USED #####
        2Sz = 0
    ioutputmode = 1          ##### DEFAULT VALUE IS USED #####

##### Print Expert input files #####

    zlocspn.def is written.
    zTrans.def is written.
    zInterAll.def is written.
    namelist.def is written.
    calcmmod.def is written.
    modpara.def is written.
    greenone.def is written.
    greentwo.def is written.

##### Input files are generated. #####

    Read File 'namelist.def'.
    Read File 'calcmmod.def' for CalcMod.
    Read File 'modpara.def' for ModPara.
    Read File 'zlocspn.def' for LocSpin.
    Read File 'zTrans.def' for Trans.
    Read File 'zInterAll.def' for InterAll.
    Read File 'greenone.def' for OneBodyG.
    Read File 'greentwo.def' for TwoBodyG.

##### Definition files are correct. #####

    Read File 'zlocspn.def'.
    Read File 'zTrans.def'.
    Read File 'zInterAll.def'.
    Read File 'greenone.def'.

```

Read File 'greentwo.def'.

Indices and Parameters of Definition files(*.def) are complete.

MAX DIMENSION idim_max=12870

APPROXIMATE REQUIRED MEMORY max_mem=0.001236 GB

MPI site separation summary

INTRA process site

Site	Bit
0	2
1	2
2	2
3	2
4	2
5	2
6	2
7	2
8	2
9	2
10	2
11	2
12	2
13	2
14	2
15	2

INTER process site

Site	Bit
------	-----

Process element info

Process	Dimension	Nup	Ndown	Nelec	Total2Sz	State
0	12870	8	8	8	0	

Total dimension : 12870

LARGE ALLOCATE FINISH !

Start: Calculate HilbertNum for fixed Sz.

End : Calculate HilbertNum for fixed Sz.

Start: Calculate diagaonal components of Hamiltonian.

```

End : Calculate diagaonal components of Hamiltonian.

##### Start: Calculate Lanczos Eigenvalue. #####

initial_mode=0 normal: iv = 6437 i_max=12870 k_exct =1

LanczosStep E[1] E[2] E[3] E[4]
stp = 2 0.7192235936 2.7807764064 xxxxxxxxxxx xxxxxxxxxx
stp = 4 -1.1878294823 0.6833997592 2.1864630296 3.4242789858
stp = 6 -2.4623912732 -0.8925857144 0.5104359160 1.7443963706
stp = 8 -3.5878334190 -2.0969913075 -0.8250723080 0.3369317092

(中略)

stp = 60 -7.1422963606 -6.8721066784 -6.6965474265 -6.5234070570
stp = 62 -7.1422963606 -6.8721066784 -6.6965474266 -6.5234070573
stp = 64 -7.1422963606 -6.8721066784 -6.6965474266 -6.5234070574

##### End : Calculate Lanczos EigenValue. #####

##### Start: Calculate Eigenvector. #####

Start: Calculate Lanczos Eigenvector.
End : Calculate Lanczos Eigenvector.
Start: Calculate Energy.
End : Calculate Energy.

Accuracy check !!!
LanczosEnergy = -7.14229636061676e+00
EnergyByVec = -7.14229636061616e+00
diff_ene = 8.50586433832080e-14
var = 9.97303843188625e-14
Accuracy of Lanczos vectors is enough.

##### End : Calculate Eigenvector. #####

Start: Calculate one body Green functions.
End : Calculate one body Green functions.

Start: Calculate two bodies Green functions.
End : Calculate two bodies Green functions.

```

この実行では、はじめにハミルトニアンの詳細を記述するファイル `zlocspin.def`、`zTrans.def`、`zInterAll.def`、`namelist.def`、`calcmmod.def`、`modpara.def` と、結果として出力する相関関数の要素を指定するファイル `greenone.def`、`greentwo.def` が生成されます。これらのファイルはエキスパートモードと共通です。

計算結果出力

Lanczos 法

Lanczos 法での計算が正常終了すると、固有エネルギーおよび一体グリーン関数、二体グリーン関数が計算され、ファイル出力されます。以下に、このサンプルでの出力ファイル例を記載します。

```
zvo_energy.dat zvo_cisajs.dat
zvo_cisajscktalt.dat
```

スタンダードモードの場合は、greenone.def、greentwo.def に基づき、一体グリーン関数には $\langle n_{i\sigma} \rangle$ 、二体グリーン関数には $\langle n_{i\sigma} n_{j\sigma'} \rangle$ が自動出力されます。なお、Lanczos 法で求めた固有ベクトルが十分な精度を持つ場合にはその固有ベクトルで計算されます。一方、Lanczos 法で求めた固有ベクトルが十分な精度を持たない場合には、ログ出力に「Accuracy of Lanczos vetor is not enough」が表示され、CG 法で固有ベクトルが求められます。各ファイルの詳細は 4.3.11, 4.3.17, 4.3.18 に記載がありますので、ご参照ください。

TPQ 法

入力ファイルで method = "TPQ" を選択すると、TPQ 法での計算が行われます。TPQ 法での計算が正常終了すると、以下のファイルが出力されます (%%には run の回数、&&には TPQ のステップ数が入ります)。

```
Norm_rand%%.dat SS_rand%%.dat
zvo_cisajs_set%%step&&.dat
zvo_cisajscktalt_set%%step&&.dat
```

Norm_rand%%.dat には、逆温度や波動関数の規格前の大きさなどの基礎情報が、各 run 回数に応じステップ数とともに出力されます。また、SS_rand%%.dat には、逆温度、エネルギー、ハミルトニアン二乗の期待値などの物理量が各 run 回数に応じステップ数とともに出力されます。zvo_cisajs_set%%step&&.dat, zvo_cisajscktalt_set%%step&&.dat には各 run 回数でのステップ数に応じた一体グリーン関数および二体グリーン関数が出力されます。各ファイルの詳細はそれぞれ、4.3.13, 4.3.14, 4.3.17, 4.3.18 に記載がありますので、ご参照ください。

全対角化法

入力ファイルで method = "fulldiag" を選択すると、全対角化法での計算が行われます。全対角化法での計算が正常終了すると、下記のファイルが出力されます (xx には 0 から始まる固有値番号が入ります)。

```
Eigenvalue.dat zvo_cisajs_eigen_xx.dat
zvo_cisajscktalt_eigen_xx.dat zvo_phys_Nup4_Ndown4.dat
```

Eigenvalue.dat には固有値番号およびエネルギー固有値が出力されます。また、zvo_cisajs_eigen_xx.dat、zvo_cisajscktalt_eigen_xx.dat には固有値番号に対応した一体グリーン関数および二体グリーン関数の値が出力されます。また、

zvo_phys_Nup4_Ndown4.dat には、エネルギーやダブロン期待値などの物理量が出力されます。各ファイルの詳細は、それぞれ 4.3.15 - 4.3.18 に記載がありますので、ご参照ください。

3.1.2 その他の系でのチュートリアル

samples/Standard/以下には次のチュートリアルが含まれています。

- 2次元正方格子 Hubbard モデル
(samples/Standard/Hubbard/square/)
- 2次元三角格子 Hubbard モデル
(samples/Standard/Hubbard/triangular/)
- 1次元近藤格子モデル
(samples/Standard/Kondo/chain/)
- 1次元反強磁性的 Heisenberg モデル
(samples/Standard/Spin/HeisenbergChain/HeisenbergChain/)
- 2次元正方格子反強磁性的 Heisenberg モデル
(samples/Standard/Spin/HeisenbergSquare/)
- Kitaev モデル (Honeycomb 格子、 2×3 サイト)
(samples/Standard/Spin/Kitaev/)

これらのチュートリアルの実行方法は前節と同じです。

3.2 エキスパートモード

エキスパートモードでは、入力ファイルとして

1. 入力ファイルリスト
2. 基本パラメータ用ファイル
3. Hamiltonian 作成用ファイル
4. 出力結果指定用ファイル

を用意した後、計算を行います。計算開始後に関しては、スタンダードモードと同様です。ここでは”*InstallDir*/samples/Expert/Spin/HeisenbergChain”にある 16 サイトの一次元鎖量子ハイゼンベルグ模型

$$H = \sum_{i=0}^{15} J \mathbf{S}_i \cdot \mathbf{S}_{i+1} \quad (3.2)$$

を例に入力ファイルの作成に関する説明を行います (ただし、 $J = 2$, $\mathbf{S}_{16} = \mathbf{S}_0$)。なお、サンプルディレクトリの中には下記の入力ファイルが用意されています。

```
calcmmod.def    greentwo.def    namelist.def    zTrans.def
greenone.def    modpara.def    zInterAll.def   zlocspn.def
```

3.2.1 入力ファイルリストファイル

入力ファイルの種類と名前を指定するファイル `namelist.def` には、下記の内容が記載されています。各入力ファイルリストファイルでは、行毎に Keyword とファイル名を記載し、ファイルの種類の区別を行います。詳細は 4.2.1 をご覧ください。

```
CalcMod calcmmod.def
ModPara modpara.def
LocSpin zlocspn.def
Trans zTrans.def
InterAll zInterAll.def
OneBodyG greenone.def
TwoBodyG greentwo.def
```

3.2.2 基本パラメータの指定

計算モード、計算用パラメータ、局在スピンの位置を以下のファイルで指定します。

計算モードの指定

CalcMod でひも付けられるファイル (ここでは `calcmmod.def`) で計算モードを指定します。ファイルの中身は下記の通りです。

```
#CalcType = 0:Lanczos, 1:TPQCalc, 2:FullDiag
#CalcMod = 0:Hubbard, 1:Spin, 2:Kondo, 3:HubbardGC,
4:SpinGC, 5:KondoGC
CalcType    0
CalcModel   1
```

CalcType で計算手法の選択、CalcModel で対象モデルの選択を行います。ここでは、計算手法として Lanczos 法、対象モデルとしてスピン系 (カノニカル) を選択しています。CalcMod ファイルでは固有ベクトルの入出力機能も指定することができます。CalcMod ファイルの詳細は 4.2.2 をご覧ください。

計算用パラメータの指定

ModPara でひも付けられるファイル (ここでは modpara.def) で計算用パラメータを指定します。ファイルの中身は下記の通りです。

```
-----
Model_Parameters    0
-----
VMC_Cal_Parameters
-----
CDataFileHead    zvo
CParaFileHead    zqp
-----
Nsite             16
Ncond             16
2Sz               0
Lanczos_max       1000
initial_iv        12
nvec              1
exct              1
LanczosEps        14
LanczosTarget     2
LargeValue        12
NumAve            5
ExpecInterval     20
```

このファイルでは、サイト数、伝導電子の数、トータル S_z や Lanczos ステップの最大数などを指定します。ModPara ファイルの詳細は 4.2.3 をご覧ください。

局在スピンの位置の指定

LocSpin でひも付けられるファイル (ここでは zlocspn.def) で局在スピンの位置と S の値を指定します。ファイルの中身は下記の通りです。


```

=====
NlocalSpin      16
=====
=====i_0LocSpn_1IteElc =====
=====
      0      1
      1      1
      2      1
      3      1
      4      1
      5      1
      ...

```

LosSpn ファイルの詳細は 4.2.4 をご覧ください。

3.2.3 Hamiltonian の指定

基本パラメータを設定した後は、Hamiltonian を構築するためのファイルを作成します。 $\mathcal{H}\Phi$ では、電子系の表現で計算を行うため、スピン系では以下の関係式

$$S_z^{(i)} = (c_{i\uparrow}^\dagger c_{i\uparrow} - c_{i\downarrow}^\dagger c_{i\downarrow})/2, \quad (3.3)$$

$$S_+^{(i)} = c_{i\uparrow}^\dagger c_{i\downarrow}, \quad (3.4)$$

$$S_-^{(i)} = c_{i\downarrow}^\dagger c_{i\uparrow} \quad (3.5)$$

を用い、電子系の演算子に変換し Hamiltonian の作成をする必要があります。

Transfer 部の指定

Trans でひも付けられるファイル (ここでは zTrans.def) で電子系の Transfer に相当する Hamiltonian

$$H_+ = - \sum_{ij\sigma_1\sigma_2} t_{ij\sigma_1\sigma_2} c_{i\sigma_1}^\dagger c_{j\sigma_2} \quad (3.6)$$

を指定します。ファイルの中身は下記の通りです。

```

=====
NTransfer        0
=====
=====i_j_s_tijs=====
=====

```

スピン系では外場を掛ける場合などに使用することができます。例えば、サイト 1 に $-0.5S_z^{(1)}$ ($S = 1/2$) の外場を掛けたい場合には、電子系の表現 $-0.5/2(c_{1\uparrow}^\dagger c_{1\uparrow} - c_{1\downarrow}^\dagger c_{1\downarrow})$ に書き換えた以下のファイルを作成することで計算することが出来ます。

```

=====
NTransfer      1
=====
=====i_j_s_tijs=====
=====
1 0 1 0 -0.25 0
1 1 1 1 0.25 0

```

Trans ファイルの詳細は 4.2.5 をご覧ください。

二体相互作用部の指定

InterAll でひも付けられるファイル (ここでは zInterAll.def) で電子系の二体相互作用部に相当する Hamiltonian

$$H+ = \sum_{i,j,k,l} \sum_{\sigma_1,\sigma_2,\sigma_3,\sigma_4} I_{ijkl\sigma_1\sigma_2\sigma_3\sigma_4} c_{i\sigma_1}^\dagger c_{j\sigma_2} c_{k\sigma_3}^\dagger c_{l\sigma_4} \quad (3.7)$$

を指定します。ファイルの中身は下記の通りです。

```

=====
NInterAll      96
=====
=====zInterAll=====
=====
0      0      0      0      1      0      1      0      0.500000  0.000000
0      0      0      0      1      1      1      1     -0.500000  0.000000
0      1      0      1      1      0      1      0     -0.500000  0.000000
0      1      0      1      1      1      1      1      0.500000  0.000000
0      0      0      1      1      1      1      0      1.000000  0.000000
0      1      0      0      1      0      1      1      1.000000  0.000000
...

```

ここでは、簡単のためサイト i とサイト $i+1$ 間の相互作用に着目して説明します。 $S = 1/2$ の場合、相互作用の項をフェルミオン演算子で書き換えると、

$$\begin{aligned}
H_{i,i+1} &= J(S_x^{(i)} S_x^{(i+1)} + S_y^{(i)} S_y^{(i+1)} + S_z^{(i)} S_z^{(i+1)}) \\
&= J \left(\frac{1}{2} S_+^{(i)} S_-^{(i+1)} + \frac{1}{2} S_-^{(i)} S_+^{(i+1)} + S_z^{(i)} S_z^{(i+1)} \right) \\
&= J \left[\frac{1}{2} c_{i\uparrow}^\dagger c_{i\downarrow} c_{i+1\downarrow}^\dagger c_{i+1\uparrow} + \frac{1}{2} c_{i\downarrow}^\dagger c_{i\uparrow} c_{i+1\uparrow}^\dagger c_{i+1\downarrow} + \frac{1}{4} (c_{i\uparrow}^\dagger c_{i\uparrow} - c_{i\downarrow}^\dagger c_{i\downarrow}) (c_{i+1\uparrow}^\dagger c_{i+1\uparrow} - c_{i+1\downarrow}^\dagger c_{i+1\downarrow}) \right]
\end{aligned}$$

となります。したがって、 $J = 2$ に対して InterAll ファイルのフォーマットを参考に相互作用を記載すると、 $S_z^{(i)} S_z^{(i+1)}$ の相互作用は

```

i      0      i      0      i+1      0      i+1      0      0.500000  0.000000
i      0      i      0      i+1      1      i+1      1     -0.500000  0.000000
i      1      i      1      i+1      0      i+1      0     -0.500000  0.000000
i      1      i      1      i+1      1      i+1      1      0.500000  0.000000

```

となり、それ以外の項は

i	0	i	1	i+1	1	i+1	0	1.000000	0.000000
i	1	i	0	i+1	0	i+1	1	1.000000	0.000000

と表せばよいことがわかります。なお、InterAll 以外にも、Hamiltonian を簡易的に記載するための下記のパイル形式に対応しています。

CoulombIntra: $n_{i\uparrow}n_{i\downarrow}$ で表される相互作用を指定します ($n_{i\sigma} = c_{i\sigma}^\dagger c_{i\sigma}$)。

CoulombInter: $n_i n_j$ で表される相互作用を指定します ($n_i = n_{i\uparrow} + n_{i\downarrow}$)。

Hund: $n_{i\uparrow}n_{j\uparrow} + n_{i\downarrow}n_{j\downarrow}$ で表される相互作用を指定します。

PairHop: $c_{i\uparrow}^\dagger c_{j\uparrow} c_{i\downarrow}^\dagger c_{j\downarrow}$ で表される相互作用を指定します。

Exchange: $S_i^+ S_j^-$ で表される相互作用を指定します。

Ising: $S_i^z S_j^z$ で表される相互作用を指定します。

PairLift: $c_{i\uparrow}^\dagger c_{i\downarrow} c_{j\uparrow}^\dagger c_{j\downarrow}$ で表される相互作用を指定します

二体相互作用に関するファイル入力形式の詳細は 4.2.6-4.2.13 をご覧ください。

3.2.4 出力ファイルの指定

一体 Green 関数および二体 Green 関数の計算する成分を、それぞれ OneBodyG, TwoBodyG でひも付けられるファイルで指定します。

一体 Green 関数の計算対象の指定

OneBodyG でひも付けられるファイル (ここでは greenone.def) で計算する一体 Green 関数 $\langle c_{i\sigma_1}^\dagger c_{j\sigma_2} \rangle$ の成分を指定します。ファイルの中身は下記の通りです。

```
=====
NCisAjs          32
=====
===== Green functions =====
=====
  0      0      0      0
  0      1      0      1
  1      0      1      0
  1      1      1      1
  2      0      2      0
  ...
```

一体 Green 関数計算対象成分の指定に関するファイル入力形式の詳細は 4.2.14 をご覧ください。

二体 Green 関数の計算対象の指定

TwoBodyG でひも付けられるファイル (ここでは greentwo.def) で計算する二体 Green 関数 $\langle c_{i\sigma_1}^\dagger c_{j\sigma_2} c_{k\sigma_3}^\dagger c_{l\sigma_4} \rangle$ の成分を指定します。ファイルの中身は下記の通りです。

```

=====
NCisAjsCktAltDC          1024
=====
===== Green functions for Sq AND Nq =====
=====
    0    0    0    0    0    0    0    0
    0    0    0    0    0    1    0    1
    0    0    0    0    1    0    1    0
    0    0    0    0    1    1    1    1
    0    0    0    0    2    0    2    0
    ...

```

二体 Green 関数計算対象成分の指定に関するファイル入力形式の詳細は 4.2.15 をご覧ください。

3.2.5 計算の実行

全ての入力ファイルが準備できた後、計算実行します。実行時はエキスパートモードを指定する“-e”をオプションとして指定の上、入力ファイルリストファイル(ここでは namelist.def)を引数とし、ターミナルから $\mathcal{H}\Phi$ を実行します。

\$ パス/HPhi -e namelist.def

計算開始後のプロセスは、スタンダードモードと同様になります。

3.2.6 その他の系でのチュートリアル

samples/Expert/以下には次のチュートリアルが含まれています。

- 2次元正方格子 Hubbard モデル
(samples/Expert/Hubbard/square/)
- 2次元三角格子 Hubbard モデル
(samples/Expert/Hubbard/triangular/)
- 1次元近藤格子モデル
(samples/Expert/Kondo/chain/)
- 1次元反強磁性的 Heisenberg モデル
(samples/Expert/Spin/HeisenbergChain/HeisenbergChain/)
- 2次元正方格子反強磁性的 Heisenberg モデル
(samples/Expert/Spin/HeisenbergSquare/)
- Kitaev モデル (Honeycomb 格子、2×3 サイト)
(samples/Expert/Spin/Kitaev/)

これらのチュートリアルの実行方法は前節と同じです。

3.3 エキスパート用入力ファイル作成モード

スタンダードモードで定義したモデルを対象に、エキスパート用入力ファイルを作成するモードです。使用方法は下記の通りです。

1. 3.1 に従い、スタンダードモデルで入力ファイルを作成します。
2. オプションとして”-sdry”を指定し、入力ファイル(ここでは StdFace.def) を記入した上で HPhi を実行します。
\$ パス/HPhi -sdry StdFace.def
3. 実行したディレクトリ内に、エキスパート用として下記の def ファイルが自動生成されます。

```
calcm0d.def    greentwo.def    namelist.def    zTrans.def  
greenone.def   modpara.def     zInterAll.def   zlocspn.def
```

4

ファイル仕様

4.1 スタンダードモード用入力ファイル

スタンダードモード用入力ファイルは次のような格好をしています。

```
W = 2
L = 4
model = "spin"
method = "Lanczos"

lattice = "triangular lattice"
//mu = 1.0
// t = -1.0
// t' = -0.5
// U = 8.0
//V = 4.0
//V'=2.0
J = -1.0
J'=-0.5
// nelec = 8
2Sz = 0
```

大まかなルールは次のとおりです。

- 各行にはひと組ずつキーワード(=の前)とパラメーター(=の後)が書かれており間は=で区切られています。
- 各キーワードは順不同に記述できます。
- 空白行、または//で始まる行(コメントアウト)は読み飛ばされます。
- 各キーワード、パラメーターの大文字・小文字は区別されません。ダブルクォート、空白は無視されます。
- 必ず指定しなければいけないパラメーター、指定しない場合デフォルト値が使われるパラメーター、(他のパラメーターの組み合わせによっては)使われないパラメーターが存在します。使われないパラメーターが指定された場合にはプログラムは終了し、入力ファイルをチェックするようにというメッセージが表示されます。

次に各キーワードの説明をします。

4.1.1 計算の種類に関する必須パラメーター

- model

形式：文字列 ("Fermion Hubbard", "Spin", "Kondo Lattice", "Fermion HubbardGC", "SpinGC" のいずれか)

説明：計算対象の模型を指定します。上記の文字列はそれぞれカノニカル集団のフェルミ粒子 Hubbard 模型

$$H = -\mu \sum_{i\sigma} c_{i\sigma}^\dagger c_{i\sigma} - \sum_{i \neq j\sigma} t_{ij} c_{i\sigma}^\dagger c_{j\sigma} + \sum_i U n_{i\uparrow} n_{i\downarrow} + \sum_{i \neq j} V_{ij} n_i n_j, \quad (4.1)$$

同じくカノニカル集団の Kitaev-Heisenberg 模型

$$H = -h \sum_i S_{iz} + \Gamma \sum_i S_{ix} + D \sum_i S_{iz} S_{iz} + \sum_{ij} (J_{ijx} S_{ix} S_{jx} + J_{ijy} S_{iy} S_{jy} + J_{ijz} S_{iz} S_{jz}), \quad (4.2)$$

カノニカル集団の近藤格子模型

$$H = -\mu \sum_{i\sigma} c_{i\sigma}^\dagger c_{i\sigma} - t \sum_{\langle ij \rangle \sigma} c_{i\sigma}^\dagger c_{j\sigma} + \frac{J}{2} \sum_i \left\{ S_i^+ c_{i\downarrow}^\dagger c_{i\uparrow} + S_i^- c_{i\uparrow}^\dagger c_{i\downarrow} + S_{iz} (n_{i\uparrow} - n_{i\downarrow}) \right\}, \quad (4.3)$$

グランドカノニカル集団のフェルミ粒子 Hubbard 模型 [式 (4.1)], グランドカノニカル集団の Kitaev-Heisenberg 模型 [式 (4.2)] に対応する。

- method

形式：文字列 ("Lanczos", "TPQ", "Full Diag" のいずれか)

説明：実行する計算の種類を指定します。上記の文字列はそれぞれランチョス法による少数固有状態の計算, 熱力学的純粋状態を用いた有限温度計算, 直接法による全固有状態計算に対応します。

- lattice

形式：文字列 ("Chain Lattice", "Square Lattice", "Triangular Lattice", "Honeycomb Lattice" のいずれか)

説明：格子の形状を指定します。上記文字列はそれぞれ1次元鎖 (Fig. 4.1(a)), 2次元正方格子 (Fig. 4.1(b)), 2次元三角格子 (Fig. 4.1(c)) に対応します。2次元異方的蜂の巣格子 (Fig. 4.2) に対応します。

4.1.2 格子に関するパラメーター

- L

形式：自然数

説明：第1次元方向の格子点数を指定します。

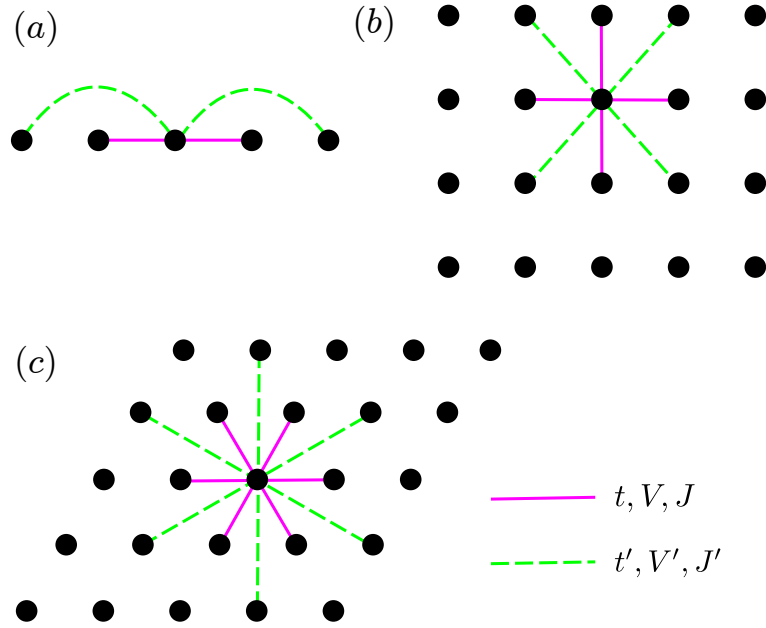


Figure 4.1: (a)1次元鎖、(b)2次元正方格子、(c)2次元三角格子の模式図. ホッピング積分、オフサイトクーロン積分、スピン結合は、再近接サイト間 (マゼンタの実線) ではそれぞれ t, V, J となり、次近接サイト間 (緑の破線) ではそれぞれ t', V', J' となります。

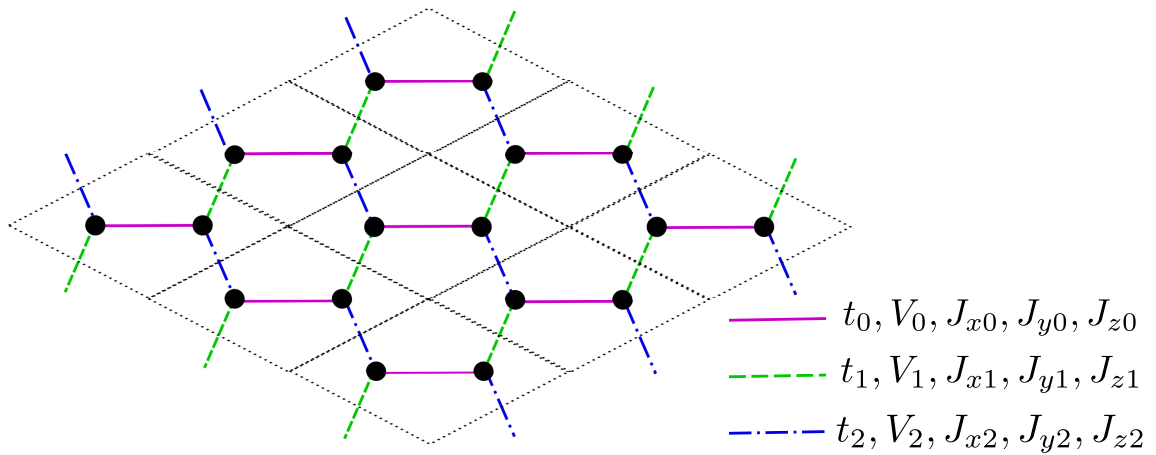


Figure 4.2: 2次元異方的蜂の巣格子の模式図. ホッピング積分、オフサイトクーロン積分、スピン結合は、ボンドの方向によって異なります。また、次近接のホッピング積分、オフサイトクーロン積分、スピン結合には対応していません。

- W

形式：自然数

説明：第2次元方向の格子点数を指定します。lattice = "Chain Lattice" の時には使われないため、指定をしないでください。

- a

形式：実数 (デフォルト値は 1.0)

説明：格子定数を指定します。

4.1.3 保存量に関するパラメーター

- nelec

形式：整数

説明：伝導電子数を指定します。lmodel = "Fermion HubbardGC", "Spin", "SpinGC" のときには指定しないでください。

- 2Sz

形式：整数

説明：全スピンのz成分の2倍を指定します。lmodel = "Fermion HubbardGC", SpinGC のときには指定しないでください。

4.1.4 Hubbard 模型パラメーター

- mu

形式：実数 (デフォルト値は 0.0)

説明：化学ポテンシャルを指定します。

- t

形式：実数 (デフォルト値は 1.0)

説明：最近接サイト間のホッピング (Fig. 4.1 参照) を指定します。

- t'

形式：実数 (デフォルト値は 0.0)

説明：次近接サイト間のホッピング (Fig. 4.1 参照) を指定します。

- t0, t1, t2

形式：実数 (デフォルト値は t)

説明：異方的蜂の巣格子での最近接サイト間のホッピング (Fig. 4.2 参照) を指定します。

- U

形式 : 実数 (デフォルト値は 0.0)

説明 : オンサイトクーロン積分を指定します。
- V, V'

形式 : 実数 (デフォルト値は 0.0)

説明 : それぞれ最近接および次近接サイト間オフサイトクーロン積分 (Fig. 4.1 参照) を指定します。
- V0, V1, V2

形式 : 実数 (デフォルト値は V)

説明 : 異方的蜂の巣格子での最近接サイト間のオフサイトクーロン積分 (Fig. 4.2 参照) を指定します。

4.1.5 Kitaev-Heisenberg 模型パラメーター

- 2S

形式 : 正の整数 (デフォルト値は 1)

説明 : 局在スピンの大きさ S の 2 倍を指定します. (例/ $1/2$ スピンならば 1)
- h, Gamma, D

形式 : 実数 (デフォルト値は 0.0)

説明 : それぞれ縦磁場、横磁場、異方性パラメータを指定します。
- J

形式 : 実数 (デフォルト値は 1.0)

説明 : 最近接サイト間の等方的スピン結合 (Fig. 4.1 参照) を指定します。これが指定されていて、 J_x, J_y, J_z, J_{xy} のいずれも指定されていない場合には J_x, J_y, J_z, J_{xy} にはすべて J が代入されます。
- Jz, Jxy

形式 : 実数 (デフォルト値は J)

説明 : 最近接サイト間の一軸性異方的スピン結合 (Fig. 4.1 参照) を指定します。 J_{xy} が指定されていて、 J_x, J_y が指定されていない場合には J_x, J_y には J_{xy} が代入されます。
- Jx, Jy

形式 : 実数 (デフォルト値は J_{xy})

説明 : 最近接サイト間の三軸性異方的スピン結合 (Fig. 4.1 参照) を指定します。

- J' , J_x' , J_y' , J_z' , J_{xy}'

形式 : 実数 (J' のデフォルト値は 0.0, J_{xy}' と J_z' のデフォルト値は J' , J_x' と J_y' のデフォルト値は J_{xy}')

説明 : 次近接サイト間のスピン結合 (Fig. 4.1 参照) を指定します。最近接のもの (J , J_x , J_y , J_z , J_{xy}) と同様のルールで設定されます。

- J_0 , J_1 , J_2 , J_{x0} , J_{y0} , J_{z0} , J_{x1} , J_{y1} , J_{z1} , J_{x2} , J_{y2} , J_{z2} , J_{xy0} , J_{xy1} , J_{xy2}

形式 : 実数 (J_0 , J_1 , J_2 のデフォルト値は J , J_{xy0} , J_{xy1} , J_{xy2} のデフォルト値は J_{xy} , J_{x0} , J_{x1} , J_{x2} のデフォルト値は J_x , J_{y0} , J_{y1} , J_{y2} のデフォルト値は J_y , J_{z0} , J_{z1} , J_{z2} のデフォルト値は J_z)

説明 : 異方的蜂の巣格子の最近接サイト間のスピン結合 (Fig. 4.2 参照) を指定します。

4.1.6 近藤格子模型パラメーター

- μ

形式 : 実数 (デフォルト値は 0.0)

説明 : 化学ポテンシャルを指定します。

- t

形式 : 実数 (デフォルト値は 1.0)

説明 : 最近接サイト間のホッピング (Fig. 4.1 参照) を指定します。

- t_0 , t_1 , t_2

形式 : 実数 (デフォルト値は t)

説明 : 異方的蜂の巣格子での最近接サイト間のホッピング (Fig. 4.2 参照) を指定します。

- J

形式 : 実数 (デフォルト値は 0.0)

説明 : 局在電子と遍歴電子のスピン結合を指定します。

4.1.7 計算条件のパラメーター

- Lanczos_max

形式 : 整数 (デフォルト値は 2000)

説明 : ランチョスステップの上限を指定します。

- initial_iv

形式 : 整数 (デフォルト値は 1)

説明 : 初期条件のベクトルを与えます。

– ランチョス法

- * カノニカル集団かつ $\text{initial_iv} \geq 0$ の場合
ノンゼロの成分が指定されます。
- * $\text{initial_iv} < 0$ の場合
乱数のシードが指定され、全ての成分に対して係数がランダムに与えられます。なお、グランドカノニカルの場合は初期状態として多くの状態を持つよう、こちらの様式が適用されます。

– TPQ 法

乱数のシードが指定され、全ての成分に対して係数がランダムに与えられます。

初期ベクトル設定の詳細については、5 を参照ください。

● nvec

形式 : 整数 (デフォルト値は 1)

説明 : ランチョス法の段階で下からいくつの固有値を求めるかを指定します。

● exct

形式 : 整数 (デフォルト値は 1)

説明 : エネルギーの低いものから数えて、何番目の固有状態を計算するかを指定します。

● LanczosEps

形式 : 整数 (デフォルト値は 14)

説明 : ランチョスの収束判定条件を指定します。ひとつ前のステップの固有値との相対誤差が、 $10^{-\text{LanczosEps}}$ 以下になったら収束したと判断します。

● LanczosTarget

形式 : 整数 (デフォルト値は 2)

説明 : エネルギーの低いものから数えて、何番目の固有値でランチョスの収束判定を行うかを指定します。

● LargeValue

形式 : 整数 (デフォルト値は下記参照)

説明 : (TPQ 法のみで使用) $l - \hat{H}/N_s$ の l 。各模型、格子ごとにデフォルト値が次のように変わります。

– Hubbard 模型 [式 (4.1)]

カノニカル集団

$$l = |\mu| \frac{N_{\text{elec}}}{N_{\text{site}}} + 2z|t| + 2z'|t'| + |U| + 2z|V| + 2z'|V'| \quad (4.4)$$

グランドカノニカル集団

$$l = 2|\mu| + 2z|t| + 2z'|t'| + |U| + 2z|V| + 2z'|V'| \quad (4.5)$$

- Kitaev-Heisenberg 模型 [式 (4.2)]

カノニカル集団

$$l = \frac{|S_z^{\text{tot}}|}{N_{\text{site}}} |h| + S|\Gamma| + S^2|D| + \frac{z}{2}S^2(|J_x| + |J_y| + |J_z|) + \frac{z'}{2}S^2(|J'_x| + |J'_y| + |J'_z|) \quad (4.6)$$

グランドカノニカル集団

$$l = S|h| + S|\Gamma| + S^2|D| + \frac{z}{2}S^2(|J_x| + |J_y| + |J_z|) + \frac{z'}{2}S^2(|J'_x| + |J'_y| + |J'_z|) \quad (4.7)$$

- 近藤格子模型 [式 (4.3)]

カノニカル集団

$$l = |\mu| \frac{N_{\text{elec}}}{N_{\text{site}}} + 2z|t| + \frac{S}{2}|J| \quad (4.8)$$

グランドカノニカル集団

$$l = 2|\mu| + 2z|t| + \frac{S}{2}|J| \quad (4.9)$$

- NumAve

形式 : 整数 (デフォルト値は 5)

説明 : (TPQ 法のみで使用) 独立な run を何回行うかを指定します。

- ExpecInterval

形式 : 整数 (デフォルト値は 20)

説明 : (TPQ 法のみで使用) 相関関数の計算を何回の TPQ ステップおきに行うかの指定。頻度を上げると計算コストが増大するので注意してください。

4.2 エキスパートモード用入力ファイル

$\mathcal{H}\Phi$ のエキスパートモードで使用する入力ファイル (*def) に関して説明します。入力ファイルの種別は以下の4つで分類されます。

(1) List:

キーワード指定なし: 使用する input file の名前のリストを書きます。なお、ファイル名は任意に指定することができます。

(2) Basic parameters:

CalcMod: 計算モードを指定するパラメーターを設定します。

ModPara: 計算時に必要な基本的なパラメーター (サイトの数、電子数、Lanczos ステップを何回やるかなど) を設定します。

LocSpin: 局在スピンの位置を設定します (近藤模型でのみ利用)。

(3) Hamiltonian: $\mathcal{H}\Phi$ の Hamiltonian を電子系の表式により指定します。

具体的には以下のファイルで指定されます。

Trans: $c_{i\sigma_1}^\dagger c_{j\sigma_2}$ で表される一体項を指定します。

InterAll: $c_{i\sigma_1}^\dagger c_{j\sigma_2} c_{k\sigma_3}^\dagger c_{l\sigma_4}$ で表される一般二体相互作用を指定します。

なお、使用頻度の高い相互作用に関しては下記のキーワードで指定することも可能です。

CoulombIntra: $n_{i\uparrow}n_{i\downarrow}$ で表される相互作用を指定します ($n_{i\sigma} = c_{i\sigma}^\dagger c_{i\sigma}$)。

CoulombInter: $n_i n_j$ で表される相互作用を指定します ($n_i = n_{i\uparrow} + n_{i\downarrow}$)。

Hund: $n_{i\uparrow}n_{j\uparrow} + n_{i\downarrow}n_{j\downarrow}$ で表される相互作用を指定します。

PairHop: $c_{i\uparrow}^\dagger c_{j\uparrow} c_{i\downarrow}^\dagger c_{j\downarrow}$ で表される相互作用を指定します。

Exchange: $S_i^+ S_j^-$ で表される相互作用を指定します。

Ising: $S_i^z S_j^z$ で表される相互作用を指定します。

PairLift: $c_{i\uparrow}^\dagger c_{i\downarrow} c_{j\uparrow}^\dagger c_{j\downarrow}$ で表される相互作用を指定します。

(4) Output:

OneBodyG : 出力する一体 Green 関数を指定します。 $\langle c_{i\sigma_1}^\dagger c_{j\sigma_2} \rangle$ が出力されます。

TwoBodyG : 出力する二体 Green 関数を指定します。 $\langle c_{i\sigma_1}^\dagger c_{j\sigma_2} c_{k\sigma_3}^\dagger c_{l\sigma_4} \rangle$ が出力されます。

4.2.1 入力ファイル指定用ファイル

計算で使用する入力ファイル一式を指定します。ファイル形式に関しては、以下ののようなフォーマットをしています。

```
CalcMod   calcmmod.def
ModPara   modpara.def
LocSpin   zlocspn.def
Trans     ztransfer.def
InterAll  zinterall.def
OneBodyG  zcisajs.def
TwoBodyG  zcisajscktaltdc.def
```

ファイル形式

[string01] [string02]

パラメータ

- [string01]
形式：string 型 (固定)
説明：キーワードを指定します。
- [string02]
形式：string 型
説明：キーワードにひも付けられるファイル名を指定します (任意)。

使用ルール

本ファイルを使用するにあたってのルールは以下の通りです。

- キーワードを記載後、半角空白を開けた後にファイル名を書きます。ファイル名は自由に設定できます。
- ファイル読込用キーワードは Table4.1 により指定します。
- 必ず指定しなければならないパラメーターは CalcMod, ModPara, LocSpin です。
- 各キーワードは順不同に記述できます。
- 指定したキーワード、ファイルが存在しない場合はエラー終了します。
- # で始まる行は読み飛ばされます。

Keywords	Details for corresponding files
CalcMod	Parameters for modes of calculation.
ModPara	Parameters for calculation.
LocSpin	Configurations of the local spins for Hamiltonian.
Trans	Transfer and chemical potential for Hamiltonian.
InterAll	Two-body interactions for Hamiltonian.
CoulombIntra	CoulombIntra interactions.
CoulombInter	CoulombInter interactions.
Hund	Hund couplings.
PairHop	Pair hopping couplings.
Exchange	Exchange couplings.
Ising	Ising interactions.
PairLift	Pair lift couplings.
OneBodyG	Output components for Green functions $\langle c_{i\sigma}^\dagger c_{j\sigma} \rangle$
TwoBodyG	Output components for Correlation functions $\langle c_{i\sigma}^\dagger c_{j\sigma} c_{k\tau}^\dagger c_{l\tau} \rangle$

Table 4.1: List of the definition files.

4.2.2 CalcMod ファイル

計算手法、計算モデル、出力モードを指定します。以下のようなフォーマットをしています。

```
CalcType    0
CalcModel   2
CalcEigenVec 0
```

ファイル形式

[string01] [int01]

パラメータ

- [string01]

形式 : string 型 (固定)

説明 : キーワードを指定します。

- [int01]

形式 : int 型

説明 : キーワードにひも付けられるパラメータを指定します。

使用ルール

本ファイルを使用するにあたってのルールは以下の通りです。

- キーワードを記載後、半角空白を開けた後に整数値を書きます。
- 各キーワードは順不同に記述できます。
- 指定したキーワード、ファイルが存在しない場合はエラー終了します。
- キーワード CalcType, CalcModel は省略できません。省略した場合はエラー終了します。
- # で始まる行は読み飛ばされます。

キーワード

次に各キーワードで指定されるパラメータに関して説明します。

- CalcType

形式 : int 型

説明 : 計算手法の指定を行います。

0: Lanczos 法

1: TPQ を利用した有限温度解析

2: 全対角化

で選択することができます。

- CalcModel

形式 : int 型

説明 : 計算モデルの指定を行います。

0: fermion Hubbard 模型 (カノニカル : 粒子数・ S_z 保存)

1: スピン模型 (カノニカル : S_z 保存)

2: 近藤模型 (カノニカル : 粒子数・ S_z 保存)

3: fermion Hubbard 模型 (グランドカノニカル : 粒子数・ S_z 非保存)

4: スピン模型 (グランドカノニカル : S_z 非保存)

5: 近藤模型 (グランドカノニカル : 粒子数・ S_z 非保存)

で選択することができます。

- CalcEigenVec

形式 : int 型 (デフォルト値 0)

説明 : 固有ベクトルを計算する際の手法の指定を行います。

0: Lanczos 法+CG 法 (Lanczos 法での収束が十分でない場合に CG 法での固有ベクトル計算が行われます)

1: Lanczos 法

で選択することができます。

- InitialVecType

形式 : int 型 (デフォルト値 0)

説明 : 固有ベクトルの初期値の型の指定を行います。

0: 複素数

1: 実数

から選択することが出来ます。

- OutputEigenVec

形式 : int 型 (デフォルト値 0)

説明 : 固有ベクトルの出力の指定を行います (MPI には非対応)。

0: 出力なし

1: 出力あり

から選択することが出来ます。

- InputEigenVec

形式 : int 型 (デフォルト値 0)

説明 : 固有ベクトルの入力の指定を行います (MPI には非対応)。

0: 入力なし

1: 入力あり

から選択することが出来ます。

4.2.3 ModPara ファイル

計算で使用するパラメータを指定します。以下のようなフォーマットをしています。

```
-----
Model_Parameters    0
-----
VMC_Cal_Parameters
-----
CDataFileHead    zvo
CParaFileHead    zqp
-----
Nsite             16
Ncond             16
2Sz               0
Lanczos_max       1000
initial_iv        12
nvec              1
exct              1
LanczosEps        14
LanczosTarget     2
LargeValue        12
NumAve            5
ExpecInterval     20
```

ファイル形式

以下のように行数に応じ異なる形式をとります。

- 1 - 5 行: ヘッダ (何が書かれても問題ありません)。
- 6 行: [string01] [string02]
- 7 - 8 行: ヘッダ (何が書かれても問題ありません)
- 9 行以降: [string01] [int01]

各項目の対応関係は以下の通りです。

- [string01]
形式 : string 型 (固定)
説明 : キーワードの指定を行います。
- [string02]
形式 : string 型 (空白不可)
説明 : アウトプットファイルのヘッダを指定します。

- [int01]

形式 : int 型 (空白不可)

説明 : キーワードでひも付けられるパラメータを指定します。

使用ルール

本ファイルを使用するにあたってのルールは以下の通りです。

- 9行目以降ではキーワードを記載後、半角空白を開けた後に整数値を書きます。
- 行数固定で読み込みを行う為、パラメータの省略はできません。

キーワード

- CDataFileHead

形式 : string 型 (空白不可)

説明 : アウトプットファイルのヘッダ。例えば、一体の Green 関数の出力ファイル名が `xxx_Lanczos_cisajs.dat` として出力されます (xxx に CDataFileHead で指定した文字が記載)。

- Nsite

形式 : int 型 (自然数)

説明 : サイト数を指定する整数。

- Ncond

Type : int 型 (自然数)

Description : 伝導電子数を指定する整数。グラントカノニカルの場合には使用されません。

- 2Sz

Type : int 型 (自然数)

Description : $2S_z$ を指定する整数。グラントカノニカルの場合には使用されません。電子系、近藤格子模型で S_z 保存を行う場合には Ncond を指定する必要があります。

- Nup

形式 : int 型 (自然数)

説明 : アップスピンの電子数を指定する整数。グラントカノニカルの場合には使用されません。

- Ndown

形式 : int 型 (自然数)

説明 : ダウンスピンの電子数を指定する整数。グラントカノニカルの場合には使用されません。

- Lanczos_max

形式 : int 型 (自然数)

説明 : Lanczos ステップを行う回数の最大値。指定した精度内で収束した場合には、これより短い回数で終了します。この回数以内に収束しない場合はエラー終了します。

- initial_iv

形式 : int 型 (整数)

説明 : 初期条件のベクトルを与えます。

- ランチョス法

- * カノニカル集団かつ $\text{initial_iv} \geq 0$ の場合
ノンゼロの成分が指定されます。

- * $\text{initial_iv} < 0$ の場合
乱数のシードが指定され、全ての成分に対して係数がランダムに与えられます。なお、グラントカノニカルの場合は初期状態として多くの状態を持つよう、こちらの様式が適用されます。

- TPQ 法

- 乱数のシードが指定され、全ての成分に対して係数がランダムに与えられます。

初期ベクトル設定の詳細については、5 を参照ください。

- nvec

形式 : int 型 (自然数)

説明 : Lanczos 法の段階で下からいくつの固有値を求めるかの数を指定する整数。

- exct

形式 : int 型 (自然数)

説明 : Lanczos 法で求める固有ベクトルの番号を指定する整数。例えば、1 なら基底状態のベクトル、2 なら第一励起状態のベクトルを求めます。 $\text{nvec} > \text{exct}$ とする必要があります。

- LanczosEps

形式 : int 型 (自然数)

説明 : Lanczos 法の収束判定条件を指定する整数。一つ前のステップの固有値との相対誤差が、 $10^{-\text{LanczosEps}}$ 以下になった場合に収束したと判定します。

- LanczosTarget

形式 : int 型 (自然数)

説明 : 何番目の固有値でランチョスの収束判定を行うかを指定する整数。1 なら基底状態。2 なら第一励起状態を示します。

- LargeValue

形式 : int 型 (自然数)

説明 : (TPQ 法のみで使用) TPQ で使用する $l - \hat{H}/N_s$ の l を指定する整数。

- NumAve

形式 : int 型 (自然数)

説明 : (TPQ 法のみで使用) 独立な run を何回行うかを指定する整数。

- ExpecInterval

形式 : int 型 (自然数)

説明 : (TPQ 法のみで使用) 相関関数の計算を何回の TPQ ステップおきに行うかの指定する整数。頻度を上げると計算コストが増大するので注意してください。

4.2.4 LocSpin 指定ファイル

局在スピンを指定します。以下のようなフォーマットをしています。

```
=====
NlocalSpin      6
=====
=====i_0LocSpn_1IteElc =====
=====
    0      1
    1      0
    2      1
    3      0
    4      1
    5      0
    6      1
    7      0
    8      1
    9      0
   10      1
   11      0
```

ファイル形式

以下のように行数に応じ異なる形式をとります。

- 1 行: ヘッダ (何かが書かれても問題ありません)。
- 2 行: [string01] [int01]
- 3-5 行: ヘッダ (何かが書かれても問題ありません)。
- 6 行以降: [int02] [int03]

パラメータ

- [string01]
形式 : string 型 (空白不可)
説明 : 局在スピンの総数を示すキーワード (任意)。
- [int01]
形式 : int 型 (空白不可)
説明 : 局在スピンの総数を指定する整数。
- [int02]
形式 : int 型 (空白不可)
説明 : サイト番号を指定する整数。0 以上 Nsite 未満で指定します。

- [int03]

形式 : int 型 (空白不可)

説明 : 局在スピンか遍歴電子かを指定する整数。

0: 遍歴電子

$n > 0$: $2S = n$ の局在スピン
を選択することが出来ます。

使用ルール

本ファイルを使用するにあたってのルールは以下の通りです。

- 行数固定で読み込みを行う為、ヘッダの省略はできません。
- [int01] と [int03] で指定される局在電子数の総数が異なる場合はエラー終了します。
- [int02] の総数が全サイト数と異なる場合はエラー終了します。^{*1}
- [int02] が全サイト数以上もしくは負の値をとる場合はエラー終了します。

^{*1}将来的には局在電子のみ指定するように変更する予定です。

4.2.5 Trans 指定ファイル

ここではハミルトニアン

$$H+ = - \sum_{ij\sigma_1\sigma_2} t_{ij\sigma_1\sigma_2} c_{i\sigma_1}^\dagger c_{j\sigma_2} \quad (4.10)$$

に対する Transfer 積分 $t_{ij\sigma_1\sigma_2}$ を指定します。以下にファイル名を記載します。

```
=====
NTransfer      24
=====
=====i_j_s_tijs=====
=====
    0      0      2      0      1.000000      0.000000
    2      0      0      0      1.000000      0.000000
    0      1      2      1      1.000000      0.000000
    2      1      0      1      1.000000      0.000000
    2      0      4      0      1.000000      0.000000
    4      0      2      0      1.000000      0.000000
    2      1      4      1      1.000000      0.000000
    4      1      2      1      1.000000      0.000000
    4      0      6      0      1.000000      0.000000
    6      0      4      0      1.000000      0.000000
    4      1      6      1      1.000000      0.000000
    6      1      4      1      1.000000      0.000000
    6      0      8      0      1.000000      0.000000
    8      0      6      0      1.000000      0.000000
...
```

ファイル形式

以下のように行数に応じ異なる形式をとります。

- 1 行: ヘッダ (何が書かれても問題ありません)。
- 2 行: [string01] [int01]
- 3-5 行: ヘッダ (何が書かれても問題ありません)。
- 6 行以降: [int02] [int03] [int04] [int05] [double01] [double02]

パラメータ

- [string01]
形式 : string 型 (空白不可)
説明 : Transfer 総数のキーワード名を指定します (任意)。

- [int01]
形式 : int 型 (空白不可)
説明 : Transfer の総数を指定します。
- [int02], [int04]
形式 : int 型 (空白不可)
説明 : サイト番号を指定する整数。0 以上 Nsite 未満で指定します。
- [int03], [int05]
形式 : int 型 (空白不可)
説明 : スピンを指定する整数。
0: アップスピン
1: ダウンスピン
を選択することが出来ます。
- [double01]
形式 : double 型 (空白不可)
説明 : $t_{ij\sigma_1\sigma_2}$ の実部を指定します。
- [double02]
形式 : double 型 (空白不可)
説明 : $t_{ij\sigma_1\sigma_2}$ の虚部を指定します。

使用ルール

本ファイルを使用するにあたってのルールは以下の通りです。

- 行数固定で読み込みを行う為、ヘッダの省略はできません。
- Hamiltonian がエルミートという制限から $t_{ij\sigma_1\sigma_2} = t_{ji\sigma_2\sigma_1}^\dagger$ の関係を満たす必要があります。上記の関係が成立しない場合にはエラー終了します。
- 成分が重複して指定された場合にはエラー終了します。
- [int01] と定義されている Transfer の総数が異なる場合はエラー終了します。
- [int02]-[int05] を指定する際、範囲外の整数を指定した場合はエラー終了します。

4.2.6 InterAll 指定ファイル

ここでは一般二体相互作用をハミルトニアンに付け加えます。付け加える項は以下で与えられます。

$$H+ = \sum_{i,j,k,l} \sum_{\sigma_1,\sigma_2,\sigma_3,\sigma_4} I_{ijkl\sigma_1\sigma_2\sigma_3\sigma_4} c_{i\sigma_1}^\dagger c_{j\sigma_2} c_{k\sigma_3}^\dagger c_{l\sigma_4} \quad (4.11)$$

なお、スピンに関して計算する場合には、 $i = j, k = l$ となるよう設定してください。以下にファイル例を記載します。

```
=====
NInterAll      36
=====
=====zInterAll=====
=====
0   0   0   1   1   1   1   0   0.50  0.0
0   1   0   0   1   0   1   1   0.50  0.0
0   0   0   0   1   0   1   0   0.25  0.0
0   0   0   0   1   1   1   1  -0.25  0.0
0   1   0   1   1   0   1   0  -0.25  0.0
0   1   0   1   1   1   1   1   0.25  0.0
2   0   2   1   3   1   3   0   0.50  0.0
2   1   2   0   3   0   3   1   0.50  0.0
2   0   2   0   3   0   3   0   0.25  0.0
2   0   2   0   3   1   3   1  -0.25  0.0
2   1   2   1   3   0   3   0  -0.25  0.0
2   1   2   1   3   1   3   1   0.25  0.0
4   0   4   1   5   1   5   0   0.50  0.0
4   1   4   0   5   0   5   1   0.50  0.0
4   0   4   0   5   0   5   0   0.25  0.0
4   0   4   0   5   1   5   1  -0.25  0.0
4   1   4   1   5   0   5   0  -0.25  0.0
4   1   4   1   5   1   5   1   0.25  0.0
...
```

ファイル形式

以下のように行数に応じ異なる形式をとります。

- 1行: ヘッダ (何が書かれても問題ありません)。
- 2行: [string01] [int01]
- 3-5行: ヘッダ (何が書かれても問題ありません)。
- 6行以降: [int02] [int03] [int04] [int05] [int06] [int07] [int08] [int09] [double01] [double02]

パラメータ

- [string01]
形式 : string 型 (空白不可)
説明 : 二体相互作用の総数のキーワード名を指定します (任意)。
- [int01]
形式 : int 型 (空白不可)
説明 : 二体相互作用の総数を指定します。
- [int02], [int04], [int06], [int08]
形式 : int 型 (空白不可)
説明 : サイト番号を指定する整数。0 以上 Nsite 未満で指定します。
- [int03], [int05], [int07], [int09]
形式 : int 型 (空白不可)
説明 : スピンを指定する整数。
0: アップスピン
1: ダウンスピン
を選択することが出来ます。
- [double01]
形式 : double 型 (空白不可)
説明 : $I_{ijkl\sigma_1\sigma_2\sigma_3\sigma_4}$ の実部を指定します。
- [double02]
形式 : double 型 (空白不可)
説明 : $I_{ijkl\sigma_1\sigma_2\sigma_3\sigma_4}$ の虚部を指定します。

使用ルール

本ファイルを使用するにあたってのルールは以下の通りです。

- 行数固定で読み込みを行う為、ヘッダの省略はできません。
- Hamiltonian がエルミートという制限から $I_{ijkl\sigma_1\sigma_2\sigma_3\sigma_4} = I_{lkji\sigma_4\sigma_3\sigma_2\sigma_1}^\dagger$ の関係を満たす必要があります。上記の関係が成立しない場合にはエラー終了します。
- スピンに関して計算する場合、 $i = j, k = l$ を満たさないペアが存在するとエラー終了します。
- 成分が重複して指定された場合にはエラー終了します。
- [int01] と定義されている InterAll の総数が異なる場合はエラー終了します。
- [int02]-[int09] を指定する際、範囲外の整数を指定した場合はエラー終了します。

4.2.7 CoulombIntra 指定ファイル

オンサイトクーロン相互作用をハミルトニアンに付け加えます ($S = 1/2$ の系でのみ使用可能、ver. 0.2 では MPI 非対応)。付け加える項は以下で与えられます。

$$H+ = \sum_i U_i n_{i\uparrow} n_{i\downarrow} \quad (4.12)$$

以下にファイル例を記載します。

```
=====
NCoulombIntra 6
=====
=====i_0LocSpn_1IteElc =====
=====
0  4.000000
1  4.000000
2  4.000000
3  4.000000
4  4.000000
5  4.000000
```

ファイル形式

以下のように行数に応じ異なる形式をとります。

- 1 行: ヘッダ (何が書かれても問題ありません)。
- 2 行: [string01] [int01]
- 3-5 行: ヘッダ (何が書かれても問題ありません)。
- 6 行以降: [int02] [double01]

パラメータ

- [string01]
形式 : string 型 (空白不可)
説明 : オンサイトクーロン相互作用の総数のキーワード名を指定します (任意)。
- [int01]
形式 : int 型 (空白不可)
説明 : オンサイトクーロン相互作用の総数を指定します。
- [int02]
形式 : int 型 (空白不可)
説明 : サイト番号を指定する整数。0 以上 Nsite 未満で指定します。

- [double01]

形式 : double 型 (空白不可)

説明 : U_i を指定します。

使用ルール

本ファイルを使用するにあたってのルールは以下の通りです。

- 行数固定で読み込みを行う為、ヘッダの省略はできません。
- 成分が重複して指定された場合にはエラー終了します。
- [int01] と定義されているオンサイトクーロン相互作用の総数が異なる場合はエラー終了します。
- [int02] を指定する際、範囲外の整数を指定した場合はエラー終了します。

4.2.8 CoulombInter 指定ファイル

オフサイトクーロン相互作用をハミルトニアンに付け加えます ($S = 1/2$ の系でのみ使用可能、ver. 0.2 では MPI 非対応)。付け加える項は以下で与えられます。

$$H+ = \sum_{i,j} V_{ij} n_i n_j \quad (4.13)$$

以下にファイル例を記載します。

```
=====
NCoulombInter 6
=====
=====CoulombInter =====
=====
  0      1 -0.125000
  1      2 -0.125000
  2      3 -0.125000
  3      4 -0.125000
  4      5 -0.125000
  5      0 -0.125000
```

ファイル形式

以下のように行数に応じ異なる形式をとります。

- 1行: ヘッダ (何が書かれても問題ありません)。
- 2行: [string01] [int01]
- 3-5行: ヘッダ (何が書かれても問題ありません)。
- 6行以降: [int02] [int03] [double01]

パラメータ

- [string01]
形式 : string 型 (空白不可)
説明 : オフサイトクーロン相互作用の総数のキーワード名を指定します (任意)。
- [int01]
形式 : int 型 (空白不可)
説明 : オフサイトクーロン相互作用の総数を指定します。
- [int02], [int03]
形式 : int 型 (空白不可)
説明 : サイト番号を指定する整数。0 以上 Nsite 未満で指定します。

- [double01]

形式 : double 型 (空白不可)

説明 : V_{ij} を指定します。

使用ルール

本ファイルを使用するにあたってのルールは以下の通りです。

- 行数固定で読み込みを行う為、ヘッダの省略はできません。
- 成分が重複して指定された場合にはエラー終了します。
- [int01] と定義されているオフサイトクーロン相互作用の総数が異なる場合はエラー終了します。
- [int02]-[int03] を指定する際、範囲外の整数を指定した場合はエラー終了します。

4.2.9 Hund 指定ファイル

Hund カップリングをハミルトニアンに付け加えます ($S = 1/2$ の系でのみ使用可能、ver. 0.2 では MPI 非対応)。付け加える項は以下で与えられます。

$$H+ = - \sum_{i,j} J_{ij}^{\text{Hund}} (n_{i\uparrow} n_{j\uparrow} + n_{i\downarrow} n_{j\downarrow}) \quad (4.14)$$

以下にファイル例を記載します。

```
=====
NHund 6
=====
=====Hund =====
=====
  0      1 -0.250000
  1      2 -0.250000
  2      3 -0.250000
  3      4 -0.250000
  4      5 -0.250000
  5      0 -0.250000
```

ファイル形式

以下のように行数に応じ異なる形式をとります。

- 1 行: ヘッダ (何が書かれても問題ありません)。
- 2 行: [string01] [int01]
- 3-5 行: ヘッダ (何が書かれても問題ありません)。
- 6 行以降: [int02] [int03] [double01]

パラメータ

- [string01]

形式 : string 型 (空白不可)

説明 : Hund カップリングの総数のキーワード名を指定します (任意)。

- [int01]

形式 : int 型 (空白不可)

説明 : Hund カップリングの総数を指定します。

- [int02], [int03]

形式 : int 型 (空白不可)

説明 : サイト番号を指定する整数。0 以上 Nsite 未満で指定します。

- [double01]

形式 : double 型 (空白不可)

説明 : J_{ij}^{Hund} を指定します。

使用ルール

本ファイルを使用するにあたってのルールは以下の通りです。

- 行数固定で読み込みを行う為、ヘッダの省略はできません。
- 成分が重複して指定された場合にはエラー終了します。
- [int01] と定義されている Hund カップリングの総数が異なる場合はエラー終了します。
- [int02]-[int03] を指定する際、範囲外の整数を指定した場合はエラー終了します。

4.2.10 PairHop 指定ファイル

PairHop カップリングをハミルトニアンに付け加えます ($S = 1/2$ の系でのみ使用可能、ver. 0.2 では MPI 非対応)。付け加える項は以下で与えられます。

$$H+ = \sum_{i,j} J_{ij}^{\text{Pair}} c_{i\uparrow}^{\dagger} c_{j\uparrow} c_{i\downarrow}^{\dagger} c_{j\downarrow} \quad (4.15)$$

以下にファイル例を記載します。

```
=====
NPairhop 6
=====
=====Pairhop =====
=====
  0      1  0.50000
  1      2  0.50000
  2      3  0.50000
  3      4  0.50000
  4      5  0.50000
  5      0  0.50000
```

ファイル形式

以下のように行数に応じ異なる形式をとります。

- 1 行: ヘッダ (何が書かれても問題ありません)。
- 2 行: [string01] [int01]
- 3-5 行: ヘッダ (何が書かれても問題ありません)。
- 6 行以降: [int02] [int03] [double01]

パラメータ

- [string01]
形式 : string 型 (空白不可)
説明 : PairHop カップリングの総数のキーワード名を指定します (任意)。
- [int01]
形式 : int 型 (空白不可)
説明 : PairHop カップリングの総数を指定します。
- [int02], [int03]
形式 : int 型 (空白不可)
説明 : サイト番号を指定する整数。0 以上 Nsite 未満で指定します。

- [double01]

形式 : double 型 (空白不可)

説明 : J_{ij}^{Pair} を指定します。

使用ルール

本ファイルを使用するにあたってのルールは以下の通りです。

- 行数固定で読み込みを行う為、ヘッダの省略はできません。
- 成分が重複して指定された場合にはエラー終了します。
- [int01] と定義されている PairHop カップリングの総数が異なる場合はエラー終了します。
- [int02]-[int03] を指定する際、範囲外の整数を指定した場合はエラー終了します。

4.2.11 Exchange 指定ファイル

Exchange カップリングをハミルトニアンに付け加えます ($S = 1/2$ の系でのみ使用可能、ver. 0.2 では MPI 非対応)。電子系の場合には

$$H+ = \sum_{i,j} J_{ij}^{\text{Ex}} (c_{i\uparrow}^\dagger c_{j\uparrow} c_{j\downarrow}^\dagger c_{i\downarrow} + c_{i\downarrow}^\dagger c_{j\downarrow} c_{j\uparrow}^\dagger c_{i\uparrow}) \quad (4.16)$$

が付け加えられ、スピン系の場合には

$$H+ = \sum_{i,j} J_{ij}^{\text{Ex}} (S_i^+ S_j^- + S_i^- S_j^+) \quad (4.17)$$

が付け加えられます。**スピン系の $(S_i^+ S_j^- + S_i^- S_j^+)$ を電子系の演算子で書き直すと、 $-(c_{i\uparrow}^\dagger c_{j\uparrow} c_{j\downarrow}^\dagger c_{i\downarrow} + c_{i\downarrow}^\dagger c_{j\downarrow} c_{j\uparrow}^\dagger c_{i\uparrow})$ となることに注意して下さい。**以下にファイル例を記載します。

```
=====
NExchange 6
=====
=====Exchange =====
=====
  0      1  0.50000
  1      2  0.50000
  2      3  0.50000
  3      4  0.50000
  4      5  0.50000
  5      0  0.50000
```

ファイル形式

以下のように行数に応じ異なる形式をとります。

- 1 行: ヘッダ (何が書かれても問題ありません)。
- 2 行: [string01] [int01]
- 3-5 行: ヘッダ (何が書かれても問題ありません)。
- 6 行以降: [int02] [int03] [double01]

パラメータ

- [string01]
形式 : string 型 (空白不可)
説明 : Exchange カップリングの総数のキーワード名を指定します (任意)。

- [int01]
形式 : int 型 (空白不可)
説明 : Exchange カップリングの総数を指定します。
- [int02], [int03]
形式 : int 型 (空白不可)
説明 : サイト番号を指定する整数。0 以上 Nsite 未満で指定します。
- [double01]
形式 : double 型 (空白不可)
説明 : J_{ij}^{Ex} を指定します。

使用ルール

本ファイルを使用するにあたってのルールは以下の通りです。

- 行数固定で読み込みを行う為、ヘッダの省略はできません。
- 成分が重複して指定された場合にはエラー終了します。
- [int01] と定義されている Exchange カップリングの総数が異なる場合はエラー終了します。
- [int02]-[int03] を指定する際、範囲外の整数を指定した場合はエラー終了します。

4.2.12 Ising 指定ファイル

Ising 相互作用をハミルトニアンに付け加えます ($S = 1/2$ の系でのみ使用可能、ver. 0.2 では MPI 非対応)。電子系の場合には

$$H+ = \sum_{i,j} J_{ij}^z (n_{i\uparrow} - n_{i\downarrow})(n_{j\uparrow} - n_{j\downarrow}) \quad (4.18)$$

が付け加えられ、スピン系の場合には

$$H+ = \sum_{i,j} J_{ij}^z S_i^z S_j^z \quad (4.19)$$

が付け加えられます。以下にファイル例を記載します。

```
=====
NIsing 6
=====
=====Ising =====
=====
  0      1  0.50000
  1      2  0.50000
  2      3  0.50000
  3      4  0.50000
  4      5  0.50000
  5      0  0.50000
```

ファイル形式

以下のように行数に応じ異なる形式をとります。

- 1 行: ヘッダ (何が書かれても問題ありません)。
- 2 行: [string01] [int01]
- 3-5 行: ヘッダ (何が書かれても問題ありません)。
- 6 行以降: [int02] [int03] [double01]

パラメータ

- [string01]

形式 : string 型 (空白不可)

説明 : Ising 相互作用の総数のキーワード名を指定します (任意)。

- [int01]

形式 : int 型 (空白不可)

説明 : Ising 相互作用の総数を指定します。

- [int02], [int03]

形式 : int 型 (空白不可)

説明 : サイト番号を指定する整数。0 以上 Nsite 未満で指定します。

- [double01]

形式 : double 型 (空白不可)

説明 : J_{ij}^z を指定します。

使用ルール

本ファイルを使用するにあたってのルールは以下の通りです。

- 行数固定で読み込みを行う為、ヘッダの省略はできません。
- 成分が重複して指定された場合にはエラー終了します。
- [int01] と定義されている Ising 相互作用の総数が異なる場合はエラー終了します。
- [int02]-[int03] を指定する際、範囲外の整数を指定した場合はエラー終了します。

4.2.13 PairLift 指定ファイル

PairLift カップリングをハミルトニアンに付け加えます ($S = 1/2$ の系でのみ使用可能、ver. 0.2 では MPI 非対応)。付け加える項は以下で与えられます。

$$H+ = \sum_{i,j} J_{ij}^{\text{PairLift}} (c_{i\uparrow}^\dagger c_{i\downarrow} c_{j\uparrow}^\dagger c_{j\downarrow} + c_{i\downarrow}^\dagger c_{i\uparrow} c_{j\downarrow}^\dagger c_{j\uparrow}) \quad (4.20)$$

以下にファイル例を記載します。

```
=====
NPairLift 6
=====
=====NPairLift =====
=====
0      1  0.50000
1      2  0.50000
2      3  0.50000
3      4  0.50000
4      5  0.50000
5      0  0.50000
```

ファイル形式

以下のように行数に応じ異なる形式をとります。

- 1行: ヘッダ (何が書かれても問題ありません)。
- 2行: [string01] [int01]
- 3-5行: ヘッダ (何が書かれても問題ありません)。
- 6行以降: [int02] [int03] [double01]

パラメータ

- [string01]
形式 : string 型 (空白不可)
説明 : PairLift カップリングの総数のキーワード名を指定します (任意)。
- [int01]
形式 : int 型 (空白不可)
説明 : PairLift カップリングの総数を指定します。
- [int02], [int03]
形式 : int 型 (空白不可)
説明 : サイト番号を指定する整数。0 以上 Nsite 未満で指定します。

- [double01]

形式 : double 型 (空白不可)

説明 : J_{ij}^{PairLift} を指定します。

使用ルール

本ファイルを使用するにあたってのルールは以下の通りです。

- スピン系のみで使用可能です。電子系、近藤系で指定した場合は計算に使用されません。
- 行数固定で読み込みを行う為、ヘッダの省略はできません。
- 成分が重複して指定された場合にはエラー終了します。
- [int01] と定義されている PairLift カップリングの総数が異なる場合はエラー終了します。
- [int02]-[int03] を指定する際、範囲外の整数を指定した場合はエラー終了します。

4.2.14 OneBodyG 指定ファイル

一体グリーン関数 $\langle c_{i\sigma_1}^\dagger c_{j\sigma_2} \rangle$ を計算します。以下にファイル例を記載します。

```
=====
NCisAjs          24
=====
===== Green functions =====
=====
  0      0      0      0
  0      1      0      1
  1      0      1      0
  1      1      1      1
  2      0      2      0
  2      1      2      1
  3      0      3      0
  3      1      3      1
  4      0      4      0
  4      1      4      1
  5      0      5      0
  5      1      5      1
  6      0      6      0
  6      1      6      1
  7      0      7      0
  7      1      7      1
  8      0      8      0
  8      1      8      1
  9      0      9      0
  9      1      9      1
 10      0     10      0
 10      1     10      1
 11      0     11      0
 11      1     11      1
```

ファイル形式

以下のように行数に応じ異なる形式をとります。

- 1行: ヘッダ (何かが書かれても問題ありません)。
- 2行: [string01] [int01]
- 3-5行: ヘッダ (何かが書かれても問題ありません)。
- 6行以降: [int02] [int03] [int04] [int05]

パラメータ

- [string01]
形式 : string 型 (空白不可)
説明 : 一体グリーン関数成分総数のキーワード名を指定します (任意)。
- [int01]
形式 : int 型 (空白不可)
説明 : 一体グリーン関数成分の総数を指定します。
- [int02], [int04]
形式 : int 型 (空白不可)
説明 : サイト番号を指定する整数。0 以上 Nsite 未満で指定します。
- [int03], [int05]
形式 : int 型 (空白不可)
説明 : スピンを指定する整数。
0: アップスピン
1: ダウンスピン
を選択することが出来ます。

使用ルール

本ファイルを使用するにあたってのルールは以下の通りです。

- 行数固定で読み込みを行う為、ヘッダの省略はできません。
- 成分が重複して指定された場合にはエラー終了します。
- [int01] と定義されている一体グリーン関数成分の総数が異なる場合はエラー終了します。
- [int02]-[int05] を指定する際、範囲外の整数を指定した場合はエラー終了します。

4.2.15 TwoBodyG 指定ファイル

二体グリーン関数 $\langle c_{i\sigma_1}^\dagger c_{j\sigma_2} c_{k\sigma_3}^\dagger c_{l\sigma_4} \rangle$ を計算します。なお、スピンに関して計算する場合には、 $i = j, k = l$ となるよう設定してください。以下にファイル例を記載します。

```
=====
NCisAjsCktAltDC          576
=====
===== Green functions for Sq AND Nq =====
=====
  0    0    0    0    0    0    0    0
  0    0    0    0    0    1    0    1
  0    0    0    0    1    0    1    0
  0    0    0    0    1    1    1    1
  0    0    0    0    2    0    2    0
  0    0    0    0    2    1    2    1
  0    0    0    0    3    0    3    0
  0    0    0    0    3    1    3    1
  0    0    0    0    4    0    4    0
  0    0    0    0    4    1    4    1
  0    0    0    0    5    0    5    0
  0    0    0    0    5    1    5    1
  0    0    0    0    6    0    6    0
  0    0    0    0    6    1    6    1
  0    0    0    0    7    0    7    0
  0    0    0    0    7    1    7    1
  0    0    0    0    8    0    8    0
  0    0    0    0    8    1    8    1
  0    0    0    0    9    0    9    0
  0    0    0    0    9    1    9    1
  0    0    0    0   10    0   10    0
  0    0    0    0   10    1   10    1
  0    0    0    0   11    0   11    0
  0    0    0    0   11    1   11    1
  0    1    0    1    0    0    0    0
  ...
```

ファイル形式

以下のように行数に応じ異なる形式をとります。

- 1行: ヘッダ (何が書かれても問題ありません)。
- 2行: [string01] [int01]
- 3-5行: ヘッダ (何が書かれても問題ありません)。

- 6 行以降: [int02] [int03] [int04] [int05] [int06] [int07] [int08] [int09]

パラメータ

- [string01]
形式 : string 型 (空白不可)
説明 : 二体グリーン関数成分総数のキーワード名を指定します (任意).
- [int01]
形式 : int 型 (空白不可)
説明 : 二体グリーン関数成分の総数を指定します.
- [int02], [int04],[int06], [int08]
形式 : int 型 (空白不可)
説明 : サイト番号を指定する整数。0 以上 Nsite 未満で指定します.
- [int03], [int05],[int07], [int09]
形式 : int 型 (空白不可)
説明 : スピンを指定する整数。
0: アップスピン
1: ダウンスピン
を選択することが出来ます.

使用ルール

本ファイルを使用するにあたってのルールは以下の通りです。

- 行数固定で読み込みを行う為、ヘッダの省略はできません。
- 成分が重複して指定された場合にはエラー終了します。
- スピンに関して計算する場合、 $i = j, k = l$ を満たさない場合ペアが存在するとエラー終了します。
- [int01] と定義されている二体グリーン関数成分の総数が異なる場合はエラー終了します。
- [int02]-[int09] を指定する際、範囲外の整数を指定した場合はエラー終了します。

4.3 出力ファイル

出力ファイルに関するファイル形式を記載します。

4.3.1 CHECK_Chemi.dat

Hamiltonian のうち化学ポテンシャル

$$H+ = \sum_{i,\sigma} \mu_{i\sigma} c_{i\sigma}^{\dagger} c_{i\sigma} \quad (4.21)$$

に関する入力確認を行います。 $\mu_{i\sigma}$ が出力されます。以下にファイル例を記載します。

```
i=0 spin=0 isite1=1 tmp_V=0.000000
i=1 spin=0 isite1=2 tmp_V=0.000000
i=2 spin=0 isite1=3 tmp_V=0.000000
i=3 spin=0 isite1=4 tmp_V=0.000000
i=4 spin=0 isite1=5 tmp_V=0.000000
i=5 spin=0 isite1=6 tmp_V=0.000000
...
```

ファイル形式

以下のようなファイル形式をとります。

- $i=[\text{int01}]$ $\text{spin}=[\text{int02}]$ $\text{isite1}=[\text{int03}]$ $\text{tmp_V}=[\text{double01}]$

パラメータ

- $[\text{int01}]$
形式 : int 型
説明 : 入力番号。
- $[\text{int02}]$
形式 : int 型
説明 : $\mu_{i\sigma}$ のスピン番号 σ を表す整数。
 0: アップスピン
 1: ダウンスピン
 を表します。
- $[\text{int03}]$
形式 : int 型
説明 : $\mu_{i\sigma}$ のサイト番号 i を表す整数。
- $[\text{double01}]$
形式 : double 型
説明 : $\mu_{i\sigma}$ の値。

4.3.2 CHECK_InterAll.dat

Hamiltonian の一般二体相互作用のうち対角成分

$$H+ = \sum_{i,j,\sigma} I_{ijj\sigma_1\sigma_1\sigma_2\sigma_2} c_{i\sigma_1}^\dagger c_{i\sigma_1} c_{i\sigma_2}^\dagger c_{i\sigma_2} \quad (4.22)$$

に関する入力確認を行います。 $I_{ijj\sigma_1\sigma_1\sigma_2\sigma_2}$ が出力されます。以下にファイル例を記載します。

```
i=0 isite1=1 A_spin=0 isite2=2 B_spin=0 tmp_V=0.500000
i=1 isite1=1 A_spin=0 isite2=2 B_spin=1 tmp_V=-0.500000
i=2 isite1=1 A_spin=1 isite2=2 B_spin=0 tmp_V=-0.500000
i=3 isite1=1 A_spin=1 isite2=2 B_spin=1 tmp_V=0.500000
i=4 isite1=2 A_spin=0 isite2=3 B_spin=0 tmp_V=0.500000
i=5 isite1=2 A_spin=0 isite2=3 B_spin=1 tmp_V=-0.500000
...
```

ファイル形式

以下のようなファイル形式をとります。

- $i=[\text{int01}]$ $\text{isite1}=[\text{int02}]$ $\text{A_spin}=[\text{int03}]$ $\text{isite2}=[\text{int04}]$ $\text{B_spin}=[\text{int05}]$ $\text{tmp_V}=[\text{double01}]$

パラメータ

- $[\text{int01}]$
形式 : int 型
説明 : 入力番号。
- $[\text{int02}], [\text{int04}]$
形式 : int 型
説明 : $I_{ijj\sigma_1\sigma_1\sigma_2\sigma_2}$ のサイト番号を表す整数。
 $[\text{int02}]$ が i , $[\text{int04}]$ が j を表します。
- $[\text{int03}], [\text{int05}]$
形式 : int 型
説明 : $I_{ijj\sigma_1\sigma_1\sigma_2\sigma_2}$ のスピン番号を表す整数。
 $[\text{int03}]$ が σ_1 , $[\text{int05}]$ が σ_2 に対応し、それぞれ
0: アップスピン
1: ダウンスピン
を表します。
- $[\text{double01}]$
形式 : double 型
説明 : $I_{ijj\sigma_1\sigma_1\sigma_2\sigma_2}$ の値。

4.3.3 CHECK_CoulombIntra.dat

Hamiltonian のオンサイトクーロン相互作用

$$H+ = \sum_i U_i n_{i\uparrow} n_{i\downarrow} \quad (4.23)$$

に関する入力確認を行います。 U_i が出力されます。以下にファイル例を記載します。

```
i=0 isite1=1 tmp_V=4.000000
i=1 isite1=2 tmp_V=4.000000
i=2 isite1=3 tmp_V=4.000000
i=3 isite1=4 tmp_V=4.000000
i=4 isite1=5 tmp_V=4.000000
i=5 isite1=6 tmp_V=4.000000
```

ファイル形式

以下のようなファイル形式をとります。

- $i=[\text{int01}]$ $\text{isite1}=[\text{int02}]$ $\text{tmp_V}=[\text{double01}]$

パラメータ

- $[\text{int01}]$
形式 : int 型
説明 : 入力番号。
- $[\text{int02}]$
形式 : int 型
説明 : U_i のサイト番号 i を表す整数。
- $[\text{double01}]$
形式 : double 型
説明 : U_i の値。

4.3.4 CHECK_Hund.dat

Hamiltonian の Hund カップリング

$$H+ = - \sum_{i,j} J_{ij}^{\text{Hund}} (n_{i\uparrow} n_{j\uparrow} + n_{i\downarrow} n_{j\downarrow}) \quad (4.24)$$

に関する入力確認を行います。 J_{ij}^{Hund} が出力されます。以下にファイル例を記載します。

```
i=0 isite1=1 isite2=2 tmp_V=0.250000
i=1 isite1=2 isite2=3 tmp_V=0.250000
i=2 isite1=3 isite2=4 tmp_V=0.250000
i=3 isite1=4 isite2=5 tmp_V=0.250000
i=4 isite1=5 isite2=6 tmp_V=0.250000
i=5 isite1=6 isite2=1 tmp_V=0.250000
```

ファイル形式

以下のようなファイル形式をとります。

- $i=[\text{int01}]$ $\text{isite1}=[\text{int02}]$ $\text{isite2}=[\text{int03}]$ $\text{tmp_V}=[\text{double01}]$

パラメータ

- $[\text{int01}]$
形式 : int 型
説明 : 入力番号。
- $[\text{int02}], [\text{int03}]$
形式 : int 型
説明 : J_{ij}^{Hund} のサイト番号を表す整数。
 $[\text{int02}]$ が i , $[\text{int03}]$ が j を表します。
- $[\text{double01}]$
形式 : double 型
説明 : J_{ij}^{Hund} の値。

4.3.5 CHECK_INTER_U.dat

Hamiltonian のオフサイトクーロン相互作用

$$H+ = \sum_i V_{ij} n_i n_j \quad (4.25)$$

に関する入力確認を行います。 V_{ij} が出力されます。以下にファイル例を記載します。

```
i=0 isite1=1 isite2=2 tmp_V=-0.125000
i=1 isite1=2 isite2=3 tmp_V=-0.125000
i=2 isite1=3 isite2=4 tmp_V=-0.125000
i=3 isite1=4 isite2=5 tmp_V=-0.125000
i=4 isite1=5 isite2=6 tmp_V=-0.125000
i=5 isite1=6 isite2=1 tmp_V=-0.125000
```

ファイル形式

以下のようなファイル形式をとります。

- $i=[\text{int01}]$ $\text{isite1}=[\text{int02}]$ $\text{isite2}=[\text{int03}]$ $\text{tmp_V}=[\text{double01}]$

パラメータ

- $[\text{int01}]$
形式 : int 型
説明 : 入力番号。
- $[\text{int02}], [\text{int03}]$
形式 : int 型
説明 : V_{ij} のサイト番号を表す整数。
 $[\text{int02}]$ が i , $[\text{int03}]$ が j を表します。
- $[\text{double01}]$
形式 : double 型
説明 : V_{ij} の値。

4.3.6 CHECK_Memory.dat

使用するメモリサイズの出力を行います。配列サイズおよび必要なメモリを出力します。以下にファイル例を記載します。

```
MAX DIMENSION idim_max=400  
REQUIRED MEMORY max_mem=0.000019 GB
```

ファイル形式

以下のようなファイル形式をとります。

- MAX DIMENSION idim_max=[int01]
- REQUIRED MEMORY max_mem =[double01] GB

パラメータ

- [int01]
形式 : int 型
説明 : ヒルベルトスペースの数を表します。
- [double01]
形式 : double 型
説明 : ヒルベルトスペースの確保に必要とするメモリサイズを表します (単位は GB)。

4.3.7 WarningOnTransfer.dat

Transfer の成分が重複している場合に出力されます。以下にファイル例を記載します。

```
double conuntings in transfers: i=0 j=2 spni 0 spnj 0
double conuntings in transfers: i=2 j=0 spni 0 spnj 0
double conuntings in transfers: i=0 j=2 spni 1 spnj 1
double conuntings in transfers: i=2 j=0 spni 1 spnj 1
```

ファイル形式

以下のようなファイル形式をとります。

- double conuntings in transfers: i=[int01] j=[int02] spni [int03] spnj [int04]

パラメータ

- [int01], [int02]

形式 : int 型

説明 : 重複している Transfer のサイト番号を表します。

- [int03], [int04]

形式 : int 型

説明 : 重複している Transfer のスピン番号を表します。

0: アップスピン

1: ダウンスピン

を表します。

4.3.8 TimeKeeper.dat

計算経過情報が出力されます。以下に Lanczos 法の場合の出力例を記載します。

```
diagonal calculation finishes: Wed Sep 16 22:58:49 2015
Lanczos Eigen Value start: Wed Sep 16 22:58:49 2015
1 th Lanczos step: Wed Sep 16 22:58:49 2015
...
122 th Lanczos step: Wed Sep 16 22:58:49 2015
Lanczos Eigenvalue finishes: Wed Sep 16 22:58:49 2015
Lanczos Eigenvector finishes: Wed Sep 16 22:58:49 2015
Lanczos expec energy finishes: Wed Sep 16 22:58:49 2015
CG Eigenvector finishes: Wed Sep 16 22:58:49 2015
CG expec energy finishes: Wed Sep 16 22:58:50 2015
CG expec_cisajs finishes: Wed Sep 16 22:58:50 2015
CG expec_cisajacktalt begins: Wed Sep 16 22:58:50 2015
```

ファイル名

- ##_TimeKeeper.dat

##は ModPara ファイル内の [string02] で指定されるヘッダを表します。

4.3.9 sz_TimeKeeper.dat

ヒルベルト空間を確保する際の経過情報が出力されます。以下に出力例を記載します。

```
initial sz : Wed Sep 16 22:58:49 2015
num_threads==4
omp parallel sz finishes: Wed Sep 16 22:58:49 2015
mid omp parallel sz : Wed Sep 16 22:58:49 2015
omp parallel sz finishes: Wed Sep 16 22:58:49 2015
```

ファイル名

- ##_sz_TimeKeeper.dat

##は ModPara ファイル内の [string02] で指定されるヘッダを表します。

4.3.10 Time_CG_EigenVector.dat

(Lanczos 法のみ) CG 法で固有ベクトルを計算する際のログを出力します。

```
allocate succeed !!!  
b[4341]=1.000000 bnorm== 1.000000  
i_itr=0 itr=5 0.0411202543 0.0000100000  
...  
i_itr=0 itr=155 0.0000000058 0.0000100000  
CG OK:   t_itr=155  
i_itr=0 itr=155 time=0.000000  
fabs(fabs(xb)-1.0)=0.9955114473313577  
b[4341]=0.004489 bnorm== 1.000000  
i_itr=1 itr=5 13.0033983157 0.0000100000  
...  
CG OK:   t_itr=275  
i_itr=1 itr=120 time=0.000000  
fabs(fabs(xb)-1.0)=0.0000000000001295  
number of iterations in inv1:i_itr=1 itr=120  
t_itr=275 0.000000
```

ファイル名

- ##_Time_CG_EigenVector.dat

##は ModPara ファイル内の [string02] で指定されるヘッダを表します。

4.3.11 energy.dat

(Lanczos 法のみ) Lanczos 法もしくは CG 法で求めた固有ベクトルを用いて計算したエネルギー、ダブロンと $\langle S_z \rangle$ を出力します。以下にファイル例を記載します。

```
Energy  -7.1043675920
Doublon 0.4164356536
Sz      0.0000000000
```

ファイル名

- ##_energy.dat

##は ModPara ファイル内の [string02] で指定されるヘッダを表します。

ファイル形式

- Energy [double01]
- Doublon [double02]
- Sz [double03]

パラメータ

- [double01]
形式 : double 型
説明 : Lanczos 法・CG 法で求めた固有ベクトルを用い計算したエネルギー。
- [double02]
形式 : double 型
説明 : Lanczos 法・CG 法で求めた固有ベクトルを用い計算したダブロン $\frac{1}{N_s} \sum_i \langle n_{i\uparrow} n_{i\downarrow} \rangle$ (N_s はサイト数)。
- [double03]
形式 : double 型
説明 : 固有ベクトルを用い計算した $\langle S_z \rangle$ 。

4.3.12 Lanczos_Step.dat

(Lanczos 法でのみ出力) Lanczos 法で固有ベクトルを求めている際のログを出力します。以下にファイル例を示します。

```
stp=4 -4.7732163164 -1.7790936582 1.2246691944 4.4823068739
stp=6 -5.8159638812 -3.6807425834 -1.3738652472 0.9326262962
stp=8 -6.2772935672 -4.8650501184 -3.1096996066 -1.1940653342
...
stp=120 -7.1043675920 -7.0817672578 -7.0646589929 -7.0008766356
stp=122 -7.1043675920 -7.0817672578 -7.0646589929 -7.0008766356
```

ファイル名

- ##_Lanczos_Step.dat

##は ModPara ファイル内の [string02] で指定されるヘッダを表します。

ファイル形式

- stp= [int01] [double01] [double02] [double03] [double04]

パラメータ

- [int01]

形式 : int 型

説明 : Lanczos 法での計算ステップ数。

- [double01], [double02], [double03], [double04]

形式 : double 型

説明 : Lanczos 法で計算時に求められた固有値。

[double01] が基底エネルギー、[double02]、[double03]、[double04] が 1, 2, 3 番目の励起エネルギーをそれぞれ表します。

4.3.13 Norm_rand.dat

(TPQ 法でのみ出力) TPQ 法での有限温度計算時のログを出力します。以下にファイル例を示します。

```
# inv_temp, global_norm, global_1st_norm, step_i
0.017471 19.046586 11.288975 1
0.034863 19.089752 11.288975 2
...
31.999572 20.802362 11.288975 1997
32.015596 20.802362 11.288975 1998
32.031620 20.802362 11.288975 1999
```

ファイル名

- Norm_rand?.?.dat

?.?は TPQ 法計算時の run の番号を表します。

ファイル形式

1 行目はヘッダで、2 行目以降は以下のファイル形式で記載されます。

- [double01] [double02] [double03] [int01]

パラメータ

- [double01]
形式 : double 型
説明 : 逆温度 $1/k_B T$ 。
- [double02]
形式 : double 型
説明 : TPQ 法で計算される規格化前の波動関数 (ベクトル) のノルム: $\langle \tilde{\psi}_k | \tilde{\psi}_k \rangle$ 。
ただし、 $|\tilde{\psi}_k\rangle \equiv (l - \hat{H}/N_s)|\psi_{k-1}\rangle$ 。
- [double03]
形式 : double 型
説明 : 規格化前の初期波動関数 (ランダムベクトル) のノルム: $\langle \tilde{\psi}_0 | \tilde{\psi}_0 \rangle$ 。ただし、 $|\tilde{\psi}_0\rangle$ は規格化前の初期波動関数。
- [int01]
形式 : int 型
説明 : 初期ランダムベクトルに $(l - \hat{H}/N_s)$ (l は ModPara ファイルの LargeValue、 N_s はサイト数) を作用させた回数。

4.3.14 SS_rand.dat

(TPQ 法でのみ出力)TPQ 法での有限温度計算結果を出力します。以下にファイル例を示します。

```
# inv_tmp, energy, phys_var, phys_doublon, phys_num, step_i
0.017471  5.526334 45.390269 1.464589 6.000000 1
0.034863  5.266718 42.655559 1.434679 6.000000 2
...
31.999572 -4.814170 23.176231 0.590568 6.000000 1997
32.015596 -4.814170 23.176231 0.590568 6.000000 1998
32.031620 -4.814170 23.176231 0.590568 6.000000 1999
```

ファイル名

- SS_rand?.dat

??は TPQ 法計算時の run の番号を表します。

ファイル形式

1 行目はヘッダで、2 行目以降は以下のファイル形式で記載されます。

- [double01] [double02] [double03] [double04] [double05] [int01]

パラメータ

- [double01]
形式 : double 型
説明 : 逆温度 $1/k_B T$ 。
- [double02]
形式 : double 型
説明 : エネルギーの期待値 $\langle H \rangle$ 。
- [double03]
形式 : double 型
説明 : ハミルトニアンの 2 乗の期待値 $\langle H^2 \rangle$ 。
- [double03]
形式 : double 型
説明 : ダブロン $\frac{1}{N_s} \sum_i \langle n_{i\uparrow} n_{i\downarrow} \rangle$ (ただし N_s はサイト数)。
- [double05]
形式 : double 型
説明 : 粒子数 $\langle \hat{n} \rangle$ 。

- [int01]

形式 : int 型

説明 : 初期ランダムベクトルに $(l - \hat{H}/N_s)$ (l は ModPara ファイルの LargeValue、 N_s はサイト数) を作用させた回数。

4.3.15 Eigenvalue.dat

(FullDiag でのみ出力) 全対角法で計算したエネルギーを出力します。以下にファイル例を記載します。

```
0 -4.8141698096
1 -3.7968502453
2 -3.2462822372
...
397 13.9898305290
398 14.4896221034
399 14.8525199079
```

ファイル形式

以下のファイル形式で記載されます。

- [int01] [double01]

パラメータ

- [int01]

形式 : int 型

説明 : 固有値の index。基底エネルギーを 0 として、エネルギーの低い順から採番する。

- [double01]

形式 : double 型

説明 : エネルギー $\langle H \rangle$ 。

4.3.16 phys.dat

(FullDiag でのみ出力) 全対角法で計算したエネルギーと物理量を出力します。エネルギーの低い基底エネルギーから順に出力されます。以下にファイル例を記載します。

<H>	<N>	<Sz>	<S2>	<D>
-4.814170	0.000000	0.000000	-0.000000	0.590568
-3.796850	0.000000	0.000000	1.333333	0.423804
...				
14.489622	0.000000	0.000000	0.000000	2.550240
14.852520	0.000000	0.000000	0.000000	2.329157

ファイル名

- カノニカル: ##_phys_Nup_\$\$Ndown%%.dat
- グランドカノニカル: ##_phys.dat

##はModPara ファイル内の [string02] で指定されるヘッダ、\$\$はNup、%%はNdownを表します。

ファイル形式

1行目はヘッダで、2行目以降は以下のファイル形式で記載されます。

- [double01] [double02] [double03] [double04] [double05]

パラメータ

- [double01]
形式 : double 型
説明 : エネルギーの期待値 $\langle H \rangle$ 。
- [double02]
形式 : double 型
説明 : 粒子数の期待値 $\langle \hat{n} \rangle$ 。
- [double03]
形式 : double 型
説明 : スピンの z 成分の期待値 $\langle S_z \rangle$ 。
- [double04]
形式 : double 型
説明 : スピンの 2 乗の期待値 $\langle S^2 \rangle$ 。

- [double05]

形式 : double 型

説明 : ダブロン $\frac{1}{N_s} \sum_i \langle n_{i\uparrow} n_{i\downarrow} \rangle$ (ただし N_s はサイト数)。

4.3.17 cisajs.dat

OneBodyG で指定された一体グリーン関数 $\langle c_{i\sigma_1}^\dagger c_{j\sigma_2} \rangle$ の計算結果を出力します。
以下にファイル例を記載します。

```

0    0    0    0 0.4452776740 0.0000000000
0    1    0    1 0.4452776740 0.0000000000
1    0    1    0 0.5000000000 0.0000000000
1    1    1    1 0.5000000000 0.0000000000
2    0    2    0 0.4452776740 0.0000000000
2    1    2    1 0.4452776740 0.0000000000
3    0    3    0 0.5000000000 0.0000000000
3    1    3    1 0.5000000000 0.0000000000
...
```

ファイル名

- Lanczos 法: ##_cisajs.dat
- TPQ 法: ##_cisajs_set??step%%.dat
- 全対角化法: ##_cisajs_eigen&&.dat

##は ModPara ファイル内の [string02] で指定されるヘッダ、??は TPQ 法計算時の run の番号、%%は TPQ 法でのステップ数、&&は固有値の番号を表します。

ファイル形式

- [int01] [int02] [int03] [int04] [double01] [double02]

パラメータ

- [int01], [int03]

形式 : int 型

説明 : サイト番号を指定する整数。[int01] が i サイト、[int03] が j サイトを表します。

- [int02], [int04]

形式 : int 型

説明 : スピンを指定する整数。[int02] が σ_1 、[int03] が σ_2 に対応します。

0: アップスピン

1: ダウンスピン

を表します。

- [double01], [double02]

形式 : double 型

説明 : $\langle c_{i\sigma_1}^\dagger c_{j\sigma_2} \rangle$ の値を表します。

[double01] が実部、[double02] が虚部を表します。

4.3.18 cisajsccktalt.dat

TwoBodyG で指定された二体グリーン関数 $\langle c_{i\sigma_1}^\dagger c_{j\sigma_2} c_{k\sigma_3}^\dagger c_{l\sigma_4} \rangle$ の計算結果を出力します。以下にファイル例を記載します。

```

0 0 0 0 0 0 0 0 0.4452776740 0.0000000000
0 0 0 0 0 1 0 1 0.1843355815 0.0000000000
0 0 0 0 1 0 1 0 0.1812412105 0.0000000000
0 0 0 0 1 1 1 1 0.2640364635 0.0000000000
0 0 0 0 2 0 2 0 0.0279690007 0.0000000000
0 0 0 0 2 1 2 1 0.2009271524 0.0000000000
0 0 0 0 3 0 3 0 0.2512810778 0.0000000000
0 0 0 0 3 1 3 1 0.1939965962 0.0000000000
...
```

ファイル名

- Lanczos 法: ##_cisajsccktalt.dat
- TPQ 法: ##_cisajsccktalt_set??step%%.dat
- 全対角化法: ##_cisajsccktalt_eigen&&.dat

##は ModPara ファイル内の [string02] で指定されるヘッダ、??は TPQ 法計算時の run の番号、%%は TPQ 法でのステップ数、&&は固有値の番号を表します。

ファイル形式

- [int01] [int02] [int03] [int04] [int05] [int06] [int07] [int08] [double01] [double02]

パラメータ

- [int01], [int03], [int05], [int07]

形式 : int 型

説明 : サイト番号を指定する整数。[int01] が i サイト、[int03] が j サイト、[int05] が k サイト、[int07] が l サイトを表します。

- [int02], [int04], [int06], [int08]

形式 : int 型

説明 : スピンを指定する整数。[int02] が σ_1 、[int04] が σ_2 、[int06] が σ_3 、[int08] が σ_4 に対応します。

0: アップスピン

1: ダウンスピン

を表します。

- [double01], [double02]

形式 : double 型

説明 : $\langle c_{i\sigma_1}^\dagger c_{j\sigma_2} c_{k\sigma_3}^\dagger c_{l\sigma_4} \rangle$ の値を表します。

[double01] が実部、[double02] が虚部を表します。

4.3.19 eigenvec.dat

CalcMod ファイルの OutputEigenVec=1 の場合に、Lanczos 法で計算された固有ベクトルを出力します。InputEigenVec=1 の場合には、出力されたファイルの読み込みを行います。以下にファイル例を記載します。

```
4900
255 0.0000896233 0.0000896690
447 -0.0000000000 -0.0000000000
495 -0.0002941096 -0.0002942596
507 -0.0000562403 -0.0000562689
510 -0.0000000000 -0.0000000000
639 -0.0000000000 0.0000000000
735 0.0002941096 0.0002942596
759 0.0000562403 0.0000562689
765 -0.0000000000 0.0000000000
831 -0.0017552852 -0.0017561802
...
```

ファイル名

- ##_eigenvec_&&.dat

##は ModPara ファイル内の [string02] で指定されるヘッダ、&&は固有値の番号を表します。

ファイル形式

- 1 行目: [int01]
- 2 行目: [int02] [double01] [double02]

パラメータ

- [int01]
形式: int 型
説明: 計算対象のヒルベルト空間数を指定する整数。
- [int02]
形式: int 型
説明: 計算対象のヒルベルト空間の数。
- [double01], [double02]
形式: double 型

説明：固有ベクトルの値を表します。
[double01] が実部、[double02] が虚部を表します。

4.4 エラーメッセージ一覧

- ERROR ! Unsupported Keyword !
存在しないキーワードを指定した場合に表示され、プログラムは停止します。
- "ERROR ! Keyword キーワード is duplicated !
同じキーワードを2回指定した場合に表示され、プログラムは停止します。
- ERROR ! Unsupported Solver : ソルバー
- ERROR ! Unsupported Model : モデル
- Sorry, this system is unsupported in the STANDARD MODE...
Please use the EXPART MODE, or write a NEW FUNCTION and post it us.
method, model, lattice のどれかまたは複数にサポートしていないパラメーターを入れた場合、プログラムは停止します。
- ERROR ! $\text{abs}(2 * \text{Sz}) > \text{nsite}$ in Hubbard model !
- ERROR ! $\text{Nelec} > 2 * \text{nsite}$ in Hubbard model !
- ERROR ! $(\text{nelec} + 2 * \text{Sz}) \% 2 \neq 0$ in Hubbard model !
- ERROR ! $\text{nelec} \leq \text{nsite} \ \&\& \ 2 * |\text{Sz}| > \text{nelec}$ in Hubbard model !
- ERROR ! $\text{nelec} > \text{nsite} \ \&\& \ 2 * |\text{Sz}| > 2 * \text{nsite} - \text{nelec}$ in Hubbard model !
- ERROR ! $\text{abs}(2 * \text{Sz}) > \text{nsite}$ in Spin model !
- ERROR ! $(\text{nsite} + 2 * \text{Sz}) \% 2 \neq 0$ in Spin model !
- ERROR ! $\text{abs}(2 * \text{Sz}) > \text{nsite}$ in Hubbard model !
- ERROR ! $\text{Nelec_cond} / 2 + \text{Nelec_loc} > \text{nsite}$ in Kondo model !
- ERROR ! $(\text{nelec_cond} + \text{nelec_loc} + 2 * \text{Sz}) \% 2 \neq 0$ in Kondo model !
- ERROR ! $\text{nelec_cond} \leq \text{nsite} / 2 \ \&\& \ 2 * |\text{Sz}| > \text{nelec_cond} + \text{nelec_loc} \dots$
- ERROR ! $\text{nelec_cond} > \text{nsite} / 2 \ \&\& \ \text{abs}(\text{Sz}^2) > \text{nsite} / 2 * 3 - \text{nelec} \dots$
カノニカル集団の計算において、入力されたサイト数、電子数、全スピン z 成分が実現できない組み合わせである場合 (例えば、電子数がサイト数の2倍よりも大きい、など) プログラムは停止します。
- Check ! キーワード is SPECIFIED but will NOT be USED.
Please COMMENT-OUT this line
or check this input is REALLY APPROPRIATE for your purpose !
使われないパラメーターを指定した時には、ユーザーに入力ファイルの確認を促しプログラムは停止します。実際に必要のないパラメーターの場合は該当する行を削除もしくはコメントアウトしてください。
- ERROR ! キーワード is NOT specified !
必ず指定しなければならないキーワードが指定されていない場合にはプログラムは停止します。
- キーワード = パラメーター ##### DEFAULT VALUE IS USED #####
これはエラーメッセージではありません。入力ファイルで指定がなかったためにデフォルト値が使われたことを知らせるメッセージです。

5

アルゴリズム

5.1 Lanczos 法

5.1.1 手法概要

Lanczos 法の解説については titpack のマニュアル [7]、線形計算の数理 [8](杉原正顕・室田一雄 著) を参考にしています。

Lanczos 法はある初期ベクトルにハミルトニアンを作用させて最大・最小近傍の固有値・固有ベクトルを求める方法です。Lanczos 法で固有値を求める際には Hilbert 空間の次元の大きさの波動関数を表すベクトルが2つ^{*1} あれば原理的には実行可能なため、大規模疎行列の対角化手法として有用であることが知られています。後述するように、固有ベクトルを求める際には Hilbert 空間の次元の大きさのベクトルがもう1本必要です。

Lanczos 法の原理はべき乗法に基づいています。べき乗法ではある任意のベクトル \mathbf{x}_0 に Hamiltonian を逐次的に作用させて、 $\hat{\mathcal{H}}^n \mathbf{x}_0$ を作成します。このとき、生成される空間 $\mathcal{K}_{n+1}(\hat{\mathcal{H}}, \mathbf{x}_0) = \{\mathbf{x}_0, \hat{\mathcal{H}}^1 \mathbf{x}_0, \dots, \hat{\mathcal{H}}^n \mathbf{x}_0\}$ は Krylov 部分空間といわれます。初期ベクトルを $\hat{\mathcal{H}}$ の固有ベクトル \mathbf{e}_i (対応する固有値を E_i とする) の重ね合わせで表すと

$$\mathbf{x}_0 = \sum_i a_i \mathbf{e}_i \quad (5.1)$$

となります。ここで、 E_0 を絶対値最大の固有値としました。またハミルトニアンはエルミートであるため、固有値は全て実数であることに注意する必要があります。これに Hamiltonian の $\hat{\mathcal{H}}^n$ を作用させると、

$$\hat{\mathcal{H}}^n \mathbf{x}_0 = E_0^n \left[a_0 \mathbf{e}_0 + \sum_{i \neq 0} \left(\frac{E_i}{E_0} \right)^n a_i \mathbf{e}_i \right] \quad (5.2)$$

となり、絶対値最大固有値 E_0 に対応する、固有ベクトルが支配的になります。適切に変換を行って、この固有ベクトルの成分を抽出するのが Lanczos 法です。

Lanczos 法では、 $\mathcal{K}_n(\hat{\mathcal{H}}, \mathbf{x}_0)$ から正規直交ベクトル $\mathbf{v}_0, \dots, \mathbf{v}_{n-1}$ を次の手続きにしたがって順次生成していきます。初期条件を $\mathbf{v}_0 = \mathbf{x}_0/|\mathbf{x}_0|$, $\beta_0 = 0$, $\mathbf{x}_{-1} = 0$ とす

^{*1}高速化のために、 $\hat{\mathcal{H}}\Phi$ ではハミルトニアンの対角成分を表すベクトル1本と、スピン z 成分 (S_z) 保存、粒子数保存の場合はその状態を指定するベクトル1本を余計に確保しています。いずれのベクトルの大きさも Hilbert 空間の次元です。

ると、正規直交基底は次の手続きによって逐次的に生成することができます。

$$\alpha_k = (\hat{\mathcal{H}}\mathbf{v}_k, \mathbf{v}_k) \quad (5.3)$$

$$\mathbf{w} = \hat{\mathcal{H}}\mathbf{v}_k - \beta_k\mathbf{v}_{k-1} - \alpha_k\mathbf{v}_k \quad (5.4)$$

$$\beta_{k+1} = |\mathbf{w}| \quad (5.5)$$

$$\mathbf{v}_{k+1} = \frac{\mathbf{v}_k}{|\mathbf{v}_k|} \quad (5.6)$$

この定義から、 α_k, β_k ともに実数であることに注意する必要があります。

これらの正規直交基底が成す Krylov 部分空間の中でもとのハミルトニアンに対する固有値問題は、

$$T_n = V_n^\dagger \hat{\mathcal{H}} V_n \quad (5.7)$$

と変形されます。ここで、 V_n を $\mathbf{v}_i (i = 0, 1, \dots, n-1)$ を並べた行列です。 T_n は三重対角行列であり、その対角成分は α_i 、副対角成分は β_i で与えられます。この三重対角行列 T_n の固有値はもとのハミルトニアンの固有値の近似値となっています ($V^\dagger V = I, I$ は単位行列であることに注意)。 T_n の固有ベクトルを $\tilde{\mathbf{e}}_i$ とするともとのハミルトニアンの固有ベクトルとの関係は $\mathbf{e}_i = V\tilde{\mathbf{e}}_i$ で与えられます。 V を覚えていれば、Lanczos 法を行うと同時に固有ベクトルを求めることができますが、実際の場合は (Hilbert 空間の次元 \times Lanczos 法の反復回数) の大きさの行列を保持することは不可能です。そこで、Lanczos 法で固有値を求めた後、 $\tilde{\mathbf{e}}_i$ を保存しておき^{*2}、再び同じ Lanczos 法の計算を行い求めた V から元の固有ベクトルを再現するようにしています。

Lanczos 法では、元の Hilbert 空間の次元より十分小さい反復回数で最大及び最小に近い固有値を精度よく求めることができることが知られています。すなわち T_n の次元 n は数百-数千程度ですみます。定量的には、最大及び最小固有値の評価の誤差は Lanczos の反復回数に対して指数関数的に減少することが示されています (詳細は Ref. [8] を参照して下さい)。

5.1.2 逆反復法

固有値の近似値がわかっているときは適当なベクトル \mathbf{y}_0 に対して $(\hat{\mathcal{H}} - E_n)^{-1}$ を逐次的に作用させれば、固有値 E_n に対応する固有ベクトルの成分が支配的になり、固有ベクトルを精度良く求めることができます。

$(\hat{\mathcal{H}} - E_n)^{-1}$ を作用させる方程式を書き換えると以下の連立方程式が得られます。

$$\mathbf{y}_k = (\hat{\mathcal{H}} - E_n)\mathbf{y}_{k+1} \quad (5.8)$$

この連立方程式を共役勾配法 (CG 法) を用いて解くことで、固有ベクトルを求めることができます。その固有ベクトルから固有値およびその他の相関関数を求めることができます。ただし、CG 法の実行にはヒルベルト空間の次元のベクトルを 4 本確保する必要があり、大規模系の計算を実行する際にはメモリが足りなくなる恐れがあるので注意が必要です。

^{*2} $\tilde{\mathbf{e}}_i$ の次元は高々 Lanczos 法の反復回数であることに注意。

5.1.3 実際の実装について

初期ベクトルの設定について

Lanczos 法では、スタンダードモード用の入力ファイル、もしくはエキスパートモードでは ModPara で指定する `initial_iv`($\equiv r_s$) により初期ベクトルの設定方法を指定します。また、初期ベクトルは ModPara で指定される入力ファイルの `InitialVecType` を用い、実数もしくは複素数の指定をすることができます。

- カノニカル集団かつ `initial_iv` ≥ 0 の場合

ヒルベルト空間のうち、

$$(N_{\text{dim}}/2 + r_s) \% N_{\text{dim}} \quad (5.9)$$

の成分が与えられます。ここで、 N_{dim} は対象となるヒルベルト空間の総数で、 $N_{\text{dim}}/2$ はデフォルト値 `initial_iv` = 1 で特殊なヒルベルト空間の選択をさけるために加えられています。なお、選択された成分の係数は実数の場合は 1、複素数の場合には $(1 + i)/\sqrt{2}$ が与えられます。

- グランドカノニカル集団 もしくは `initial_iv` < 0 の場合

初期ベクトルはランダムベクトルとして与えられます。乱数のシードは

$$123432 + |r_s| \quad (5.10)$$

で指定します。ここで、 n_{run} は run の回数であり、run の最大回数はスタンダードモード用入力ファイル、もしくは ModPara で指定される入力ファイルの `NumAve` で指定します。`initial_iv` は入力ファイルで指定します。乱数は SIMD-oriented Fast Mersenne Twister (dSFMT) を用い発生させています [?]

収束判定について

$\mathcal{H}\Phi$ では、 T_n の対角化に lapack のルーチン `dsyev` を使用しており、 T_n の基底状態の次の固有値（第一励起状態のエネルギー）を収束判定条件に用いています。デフォルトの設定では、最初の 5 回の Lanczos ステップの後に、2 回毎に T_n の対角化を行い、前の Lanczos ステップの第一励起状態のエネルギーと指定した精度以内で一致すれば、収束したと判定しています。なお、収束する際の精度は `CDataFileHead` (エキスパートモードでは ModPara ファイル内) で指定することが可能です。

その後、Lanczos 法を再度行い、逐次 V を求めて、指定した準位の固有ベクトルを求めます。得られた固有ベクトル $|n\rangle$ を用い、エネルギーの期待値 $E_n = \langle n | \hat{\mathcal{H}} | n \rangle$ 、およびバリエーション $\Delta = \langle n | \hat{\mathcal{H}}^2 | n \rangle - (\langle n | \hat{\mathcal{H}} | n \rangle)^2$ を求めて、 E_n が Lanczos 法で求めた固有値と指定した精度で一致しているか、バリエーションが指定した精度以下になっているかをチェックしています。指定した精度に達していれば、対角化を終了しています。

指定した精度に達していない場合には逆反復法を用いて再度固有ベクトルを求め直します。逆反復法の初期ベクトルとして Lanczos 法で求めた固有ベクトルをとった方が一般に収束が早いので、標準の設定ではそのように取っています。

5.2 完全対角化

5.2.1 手法概要

ハミルトニアン \hat{H} を実空間配置 $|\psi_j\rangle (j = 1 \cdots N)$ を用いて作成します: $H_{ij} = \langle \psi_i | \hat{H} | \psi_j \rangle$ 。この行列を対角化することで、固有値 E_i 、固有ベクトル $|\Phi_i\rangle$ を求めることができます ($i = 1 \cdots N$)。なお、対角化では lapack の **dsyev** また **zheev** を用いています。また、有限温度計算用に各固有エネルギー状態の期待値 $\langle A_i \rangle \equiv \langle \Phi_i | \hat{A} | \Phi_i \rangle$ を計算・出力するようにしています。

5.2.2 有限温度物理量の計算

完全対角化で求めた $\langle A_i \rangle \equiv \langle \Phi_i | \hat{A} | \Phi_i \rangle$ を用い、

$$\langle \hat{A} \rangle = \frac{\sum_{i=1}^N \langle A_i \rangle e^{-\beta E_i}}{\sum_{i=1}^N e^{-\beta E_i}} \quad (5.11)$$

の関係から有限温度の物理量を計算します。実際の計算処理としてはポスト処理により計算を行います。実際の使用方法に関しては、2 を参照して下さい。

5.3 熱的純粋量子状態による有限温度計算

杉浦・清水によって、少数個（サイズが大きい場合はほぼ一つ）の波動関数から有限温度の物理量を計算する方法が提案されました [5]。その状態は熱的純粋量子状態 (TPQ) と呼ばれています。TPQ はハミルトニアンを波動関数に順次作用させて得られるので、Lanczos 法の技術がそのまま使うことができます。ここでは、とくに計算が簡単な、micro canonical TPQ (mTPQ) の概要を述べます。

$|\psi_0\rangle$ をあるランダムベクトルとします。これに $(l - \hat{H}/N_s)(l$ はある定数、 N_s はサイト数) を k 回作用させた（規格化された）ベクトルは次のように与えられます。

$$|\psi_k\rangle \equiv \frac{(l - \hat{H}/N_s)|\psi_{k-1}\rangle}{|(l - \hat{H}/N_s)|\psi_{k-1}\rangle|}. \quad (5.12)$$

この $|\psi_k\rangle$ が mTPQ 状態で、この mTPQ 状態に対応する逆温度 β_k は以下のように内部エネルギー u_k から求めることができます。

$$\beta_k \sim \frac{2k/N_s}{l - u_k}, \quad u_k = \langle \psi_k | \hat{H} | \psi_k \rangle / N_s. \quad (5.13)$$

そして、任意^{*3}の物理量 \hat{A} の β_k での平均値は

$$\langle \hat{A} \rangle_{\beta_k} = \langle \psi_k | \hat{A} | \psi_k \rangle / N_s \quad (5.14)$$

となります。有限系では最初の乱数ベクトルによる誤差がありますので、いくつか独立な計算を行って、 $|\psi_0\rangle$ に関する平均値および標準偏差を見積もっています。

^{*3}局所的という条件がつきます。

5.3.1 実際の実装について

初期ベクトルの設定について

熱的純粋量子状態による有限温度計算では、初期ベクトルは全ての成分に対してランダムな係数を与えます。初期ベクトルの係数の型は `ModPara` で指定される入力ファイルの `InitialVecType` を用い、実数もしくは複素数の指定をすることができます。乱数のシードは `initial_iv` ($\equiv r_s$) により

$$123432 + (n_{\text{run}} + 1) \times |r_s| \quad (5.15)$$

で与えられます。ここで、 n_{run} は run の回数であり、run の最大回数はスタンダードモード用入力ファイル、もしくは `ModPara` で指定される入力ファイルの `NumAve` で指定します。`initial_iv` はスタンダードモード用の入力ファイル、もしくはエキスパートモードでは `ModPara` で指定される入力ファイルで指定します。乱数は SIMD-oriented Fast Mersenne Twister (dSFMT) を用い発生させています [?]

6

謝辞

$\mathcal{H}\Phi$ の開発に当たり、西森秀稔教授が開発された先駆的な数値対角化パッケージ TITPACK [7] の実装を参考にしました。特にランチョス法の部分は TITPACK の fortran コードを C 言語に移植したものがもとになっています。また、 $\mathcal{H}\Phi$ のエキスパートモードにおけるユーザー・インターフェースの設計の際、田原大資氏が開発された mVMC の柔軟なユーザー・インターフェースが礎となっています。インターフェースに関するコードの一部分は mVMC のものを使用しています。この場を借りて、お二人に感謝します。

$\mathcal{H}\Phi$ ver.0.1, ver.0.2 は、2015 年度 東京大学物性研究所 ソフトウェア高度化プロジェクトの支援を受け開発されました。この場を借りて感謝します。

References

- [1] E. Dagotto, Rev. Mod. Phys. **66**, 763–840 (1994).
- [2] M. Imada, M. Takahashi, Journal of the Physical Society of Japan **55**, 3354–3361 (1986).
- [3] J. Jaklič, P. Prelovšek, Phys. Rev. B **49**, 5065–5068 (1994).
- [4] A. Hams, H. De Raedt, Phys. Rev. E **62**, 4365–4377 (2000).
- [5] S. Sugiura, A. Shimizu, Phys. Rev. Lett. **108**, 240401 (2012).
- [6] Y. Yamaji, Y. Nomura, M. Kurita, R. Arita, M. Imada, Phys. Rev. Lett. **113**, 107201 (2014).
- [7] http://www.stat.phys.titech.ac.jp/nishimori/titpack2_new/index-e.html.
- [8] 杉原正顯, 室田一雄, 線形計算の数理, 岩波数学叢書, 岩波書店, 2009.