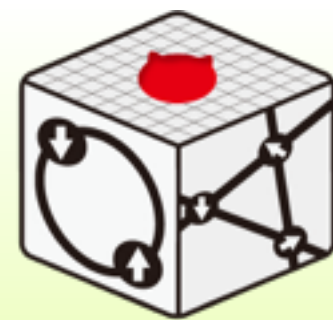


mVMCの演習問題

三澤 貴宏

東京大学物性研究所 特任研究員 (PCoMS PI)



mVMC

基本

1. Heisenberg, Hubbard chain
2. Heisenberg, Hubbard 正方格子
3. 補助ツールI: fourier tool の使用
4. 補助ツールII: UHFを初期解にする

`mVMC-tutorial/HandsOn/2017_0830/Samples`
以下にスクリプト例があります

発展 (mVMCで様々な量子相を作ってみましょう)

1. Hubbard + $V \rightarrow$ 電荷秩序
2. Heisenberg+J2 \rightarrow ストライプ磁気秩序
3. Attractive Hubbard \rightarrow 超伝導
4. 1D Kondo lattice \rightarrow 近藤絶縁体
5. Kitaev model \rightarrow Kitaev spin 液体

1. Heisenberg chain

$$H = \sum_{\langle i,j \rangle} \vec{S}_i \cdot \vec{S}_j$$

mVMC-tutorial/HandsOn/2017_0830/Samples/1D_Heisenberg

step 1. L=4でmVMCで最適化計算をしてみましょう

StdFace.defの例

```
L = 4
Lsub = 2
model = "Spin"
lattice = "chain"
J = 1.0
2Sz = 0
NVMCSample = 200
NSROptItrStep = 500
NSROptItrSmp = 50
NMPTrans = 1
NSPStot = 0
```

実行例

1. ./vmc.out -s StdFace.def

2. gnuplotで

plot ./output/zvo_out_001.dat u 1

Energy by $H\Phi$

```
0 -2.00000000000 : S=0
1 -1.00000000000 : S=1
2 -0.00000000000
3 0.00000000000
4 0.00000000000
5 1.00000000000
```

1. Heisenberg chain

$$H = \sum_{\langle i,j \rangle} \vec{S}_i \cdot \vec{S}_j$$

step 2. L=4でmVMCで相関関数計算をしてみましょう

StdFace_aft.def

```
L = 4
Lsub = 2
model = "Spin"
lattice = "chain"
J = 1.0
2Sz = 0
NVMCSample = 200
NVMCCalMode = 1
NDataIdxStart = 1
NDataQtySmp = 5
NMPTrans = 1
NSPStot = 0
```

実行例

1. `cp ./output/zqp_opt.dat .`
2. `./vmc.out -s StdFace_aft.def ./zqp_opt.dat`

1. Heisenberg chain

$$H = \sum_{\langle i,j \rangle} \vec{S}_i \cdot \vec{S}_j$$

step 3. 物理量の平均値・標準誤差を計算してみましょう

output/zvo_out_001.dat
output/zvo_out_002.dat
output/zvo_out_003.dat
output/zvo_out_004.dat
output/zvo_out_005.dat

→独立なbinでのエネルギーの
計算の値(一列目)

この平均値・標準誤差を
計算すればよい

```
ln -s output aft
perl -w Aft_energy.pl
```

で計算できます。

同様に

output/zvo_cisajs_00n.dat
output/zvo_cisajsckalt_00n.dat
に独立なbinでの

1体・2体の相関関数の値が出力

この平均値・標準誤差を計算すればよい

```
ln -s output aft
perl -w Aft_Sq.pl
```

でスピン構造因子が計算できます。

注: Result_Sq.datの1列目は

$Lx \cdot kx / (2\pi)$ です。

1. Heisenberg chain

$$H = \sum_{\langle i,j \rangle} \vec{S}_i \cdot \vec{S}_j$$

mVMC-tutorial/HandsOn/2017_0830/Samples/1D_Heisenberg

step 4.以上のことを一括して行なうのが X.sh です

sh ./X.sh

で構造因子の計算まで全て終わります

[必要な実行体(vmc.out, vmcdry.out)は計算するディレクトリにおいておく必要があります]

input.txtをいじるとサイズ・total spin(NSPStot)が変更られます

```
Lx 4
Ly 1
orb_num 1
NSPStot 0
```

- Result_Sq.dat をプロットしてみましょう
- サイズをいくつか変えて計算してみましょう
- S=1にして計算して励起状態が計算できるか試してみましょう

[$S(q=0) = S(S+1)/(3L^2)$ になっているか?]

Scriptの説明

sh ./X.sh → 全ての計算を行なう

sh ./Clean.sh → 全てのdefファイルなどを消去(初期化)

[計算結果は全て消えるので注意!!!]

perl -w MakeMod.pl : input.txtから,
StdFace.def (最適化1,結果はopt1),
StdFace_2.def (最適化2, 結果はopt2),
StdFace_aft.def (物理量計算用,結果はaft)
を作成します

perl -w CisAjs.pl : 1体の相関関数のインプットファイルを作成

perl -w CisAjsCktAltDC.pl: 2体の相関関数のインプットファイルを作成

perl -w Aft_Sq.pl : aft/の相関関数から構造因子を計算します

perl -w Aft_energy.pl : aft/のエネルギーから期待値・標準誤差

perl -w Aft_SiSj.pl : aft/ の相関関数から最近接のスピン相関を計算します

1. Heisenberg chain (references)

L=6:

0	-2.8027756377
1	-2.1180339887
2	-1.5000000000
3	-1.2807764064
4	-1.2807764064
5	-1.0000000000
6	-1.0000000000
7	-0.5000000000

L=8:

0	-3.6510934089
1	-3.1284190638
2	-2.6996281483
3	-2.4587385089
4	-2.4587385089
5	-2.1451483739
6	-2.1451483739
7	-1.8546376797

L=10:

0	-4.5154463545
1	-4.0922073467
2	-3.7705974354
3	-3.5432793743
4	-3.5432793743
5	-3.2461649167
6	-3.2461649167
7	-2.9759318691

1. Hubbard chain

$$H = -t \sum_{\langle i,j \rangle} (c_{i\sigma}^\dagger c_{j\sigma} + \text{h.c.}) + U \sum_i n_{i\uparrow} n_{i\downarrow}$$

Step.1 L=4, U=4, t=1, half fillingで計算をしてみましょう.

StdFace.defの例

```
L = 4
Lsub = 2
model = "FermionHubbard"
lattice = "chain"
t = 1.0
U = 4.0
nelec = 4
2Sz = 0
NVMCSample = 200
NSROptItrStep = 500
NSROptItrSmp = 50
NMPTrans = 1
```

Energy by HΦ

0	-2.1027484835
1	-1.8064238518
2	-1.0681403934
3	-0.8284271247
4	-0.8284271247
5	0.0000000000
6	0.5814492811
7	2.0000000000

1. Hubbard chain

$$H = -t \sum_{\langle i,j \rangle} (c_{i\sigma}^\dagger c_{j\sigma} + \text{h.c.}) + U \sum_i n_{i\uparrow} n_{i\downarrow}$$

Step. 2 物理量計算をやってみましょう

Step. 3 物理量の平均値・標準誤差を計算してみましょう

Step. 4 一括スクリプトで計算してみましょう(X.sh)

U, 電子数を変えるなどして色々計算してみましょう

HΦの計算結果と比べて見ましょう

HΦのスクリプト例

`./HPhi -s StdFace.def`

```
L           = 4
model       = "Hubbard"
lattice     = "chain"
method      = "fulldiag"
U           = 4.0
t           = 1.0
2Sz         = 0
nelec       = 4
```

```
L           = 8
model       = "Hubbard"
lattice     = "chain"
method      = "CG"
U           = 4.0
t           = 1.0
2Sz         = 0
nelec       = 8
exct        = 8
```

1. Hubbard chain (references)

HΦの結果(全対角化)

L=6:

0	-3.6687061788729571
1	-2.8983814740367304
2	-2.5163768731161431
3	-2.4229112638479289
4	-2.4229112638479293
5	-2.0927538294969210
6	-2.0927538294969210
7	-1.7690248232884345

L=8:

0	-4.6035262999892002
1	-4.2999927584330599
2	-4.0101539576440342
3	-3.7057642394839405
4	-3.7057642394839405
5	-3.4963563102152051
6	-3.4963563102152042
7	-3.2445570984649694

L=6, 8では運動量射影もできたらやってみましょう.

NMPTrans = 2などと指定.

2. Heisenberg & Hubbard on the square lattice

- chainの場合とやることは同じです。計算が重くなるので、 $4 \times 4 = 16$ sites程度にしておいた方がよいです。
- `sp Result_Sq.dat` or `fourier tool` を使うと構造因子の3次元プロットがでます。

2. Heisenberg & Hubbard on the square lattice

Heisenberg model

$(N_s = 4 \times 4)$	E/N_s	S_{nn}	S_{nnn}	$S(\mathbf{q}_{\text{peak}})$
ED	-0.70178020	-0.35089010	0.21376	0.09217
mVMC(2×2)	-0.701769(6)	-0.35088(3)	0.2136(2)	0.09212(6)
mVMC(2×2)+Lanczos	-0.701783(3)	-	-	-
mVMC(4×4)	-0.70178015(8)	0.35089007(4)	0.2139(4)	0.0922(1)
$(N_s = 6 \times 6)$	E/N_s	S_{nn}	S_{nnn}	$S(\mathbf{q}_{\text{peak}})$
ED	-0.6788721499	-0.33943607	0.207402499	0.069945
mVMC(2×2)	-0.67843(2)	-0.33921(1)	0.20738(1)	0.07019(4)
mVMC(2×2)+Lanczos	-	-	-	-
mVMC(6×6)	-0.678865(5)	-0.339433(3)	0.2072(2)	0.0698(1)
mVMC(6×6)+Lanczos	-0.678881(5)	-	-	-

Table 4: Comparisons with exact diagonalization for 4×4 and 6×6 Heisenberg model with $J = 1$. We note $\mathbf{q}_{\text{peak}} = (\pi, \pi)$. The relative errors η become 0.000001% for $L = 4$ and 0.001% for $L = 6$, respectively.

2. Heisenberg & Hubbard on the square lattice

Hubbard model

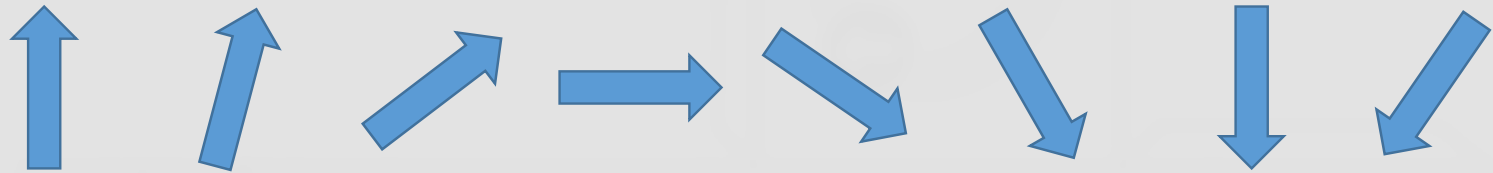
	E/N_s	D	S_{nn}	$S(\mathbf{q}_{\text{peak}})$
ED	-0.85136	0.11512	-0.2063	0.05699
mVMC(2×2)	-0.84982(4)	0.11529(5)	-0.2062(1)	0.05773(4)
mVMC(2×2)+Lanczos	-0.85105(3)	-	-	-
mVMC(4×4)	-0.85068(4)	0.1153(4)	-0.2062(5)	0.0573(2)
mVMC(4×4)+Lanczos	-0.85121(3)	-	-	-

Table 3: Comparisons with exact diagonalization for 4×4 Hubbard model with $U = 4$ and $t = 1$ at half filling. Exact diagonalization (ED) is done by using $\mathcal{H}\Phi$ [38, 39]. mVMC(2×2) means f_{ij} has 2×2 sublattice structures, $\mathbf{q}_{\text{peak}} = (\pi, \pi)$, and the parentheses denote the error bars in the last digit. Lanczos means that the first-step Lanczos calculations on top of the mVMC calculations.

3. fourier tool

相関関数のフーリエ変換

長距離の相関を調べる。



$$\langle \hat{A}_k^\dagger A_k \rangle = \frac{1}{N_{\text{Cell}}} \sum_{ij}^{N_{\text{site}}} e^{i\mathbf{k} \cdot (\mathbf{R}_i - \mathbf{R}_j)} \langle \hat{A}_i^\dagger A_j \rangle$$

ユーティリティ・プログラムとドキュメントはmVMC本体と別にある。

tool/fourier : フーリエ変換をするプログラム

tool/corplot : 3次元プロットをするプログラム

doc/userguid.html : からマニュアルを閲覧できる

計算結果

例/正方格子ハイゼンベルグ模型(16サイト)

sample/Standard/Spin/HeisenbergSquare/

```
$ パス/vmc.out -s StdFace.def
$ パス/vmc.out -s StdFace.def output/zqp_opt.dat
$ パス/fourier namelist.def geometry.dat
$ パス/corplot output/zvo_corr.dat
```

```
L = 4
W = 4
Lsub = 2
Wsub = 2
model = "Spin"
lattice = "tetragonal"
J = 1.0
NSROptItrStep = 200
2Sz = 0
NVMCCalMode = 1 [コメント(//)を外す]
```

```
##### Plot Start #####
```

Please specify target number from below (0 or Ctrl-C to exit):

Real Part Without ErrorBar

[1] Up-Up [2] Down-Down [3] Density-Density [4] SzSz [5] S+S- [6] S.S

Imaginary Part Without ErrorBar

[11] Up-Up [12] Down-Down [13] Density-Density [14] SzSz [15] S+S- [16] S.S

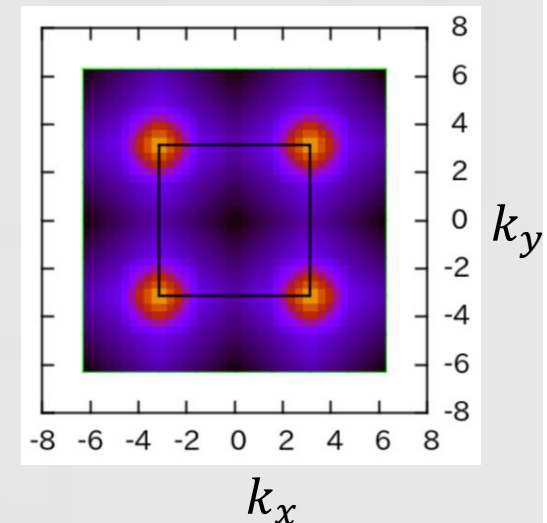
Real Part With ErrorBar

[21] Up-Up [22] Down-Down [23] Density-Density [24] SzSz [25] S+S- [26] S.S

Imaginary Part With ErrorBar

[31] Up-Up [32] Down-Down [33] Density-Density [34] SzSz [35] S+S- [36] S.S

Target : 6 (と打ってEnter)



既知の問題点

corplot内でgnuplotを呼び出しているが、
4.4より前のバージョンのgnuplotでは描画できない。

注意: fourier toolとperl スクリプトの違い

perl スクリプト (Aft_Sq.pl)

$$S(q) = \frac{1}{3N_s^2} \sum_{\langle i,j \rangle} \langle \vec{S}_i \vec{S}_j \rangle e^{iq(r_i - r_j)}$$

$$N(q) = \frac{1}{N_s^2} \sum_{\langle i,j \rangle} \langle (\vec{N}_i - \langle \vec{N}_i \rangle) \cdot (\vec{N}_j - \langle \vec{N}_j \rangle) \rangle e^{iq(r_i - r_j)}$$

fourier tool

$$S(q) = \frac{1}{N_s} \sum_{\langle i,j \rangle} \langle \vec{S}_i \vec{S}_j \rangle e^{iq(r_i - r_j)}$$

$$N(q) = \frac{1}{N_s} \sum_{\langle i,j \rangle} \langle \vec{N}_i \vec{N}_j \rangle e^{iq(r_i - r_j)}$$

4. UHF解を初期条件にする

UHF解を初期条件にしてみましょう UHFの実行体は以下にあります

/src/ComplexUHF/UHF

./UHF namelist.def

で計算可能

zqp_AP0rbital_opt.dat (fijの初期値)

が生成されます

注意: 初期条件を適切につくる必要があります

IniGreen.pl → 正方格子(π, π)の磁気秩序を初期条件

namelist.def に Initial zinitial.def を追加

4. UHF解を初期条件にする

```
sh ./IniUHF.sh
```

で平均場解からfijが生成される（UHFは実行するディレクトリにおいておく）

IniUHF.shの中身

```
perl -w MakeMod.pl
./vmcdry.out StdFace.def
#[s]UHF
mkdir tmpUHF
cp IniGreen.pl ./tmpUHF
cp *def ./tmpUHF
cp input.txt ./tmpUHF
cp ./UHF ./tmpUHF
cd ./tmpUHF
perl -w IniGreen.pl
echo "          Initial zinitial.def" >> namelist.def
./UHF namelist.def
cd -
#[e]UHF
cp tmpUHF/zqp_AP0orbital_opt.dat .
echo "          InOrbital zqp_AP0orbital_opt.dat" >> namelist.def
cp namelist.def xnamelist.def
```

X.sh の最初の最適化のnamelitst.defをxnamelist.defに変更

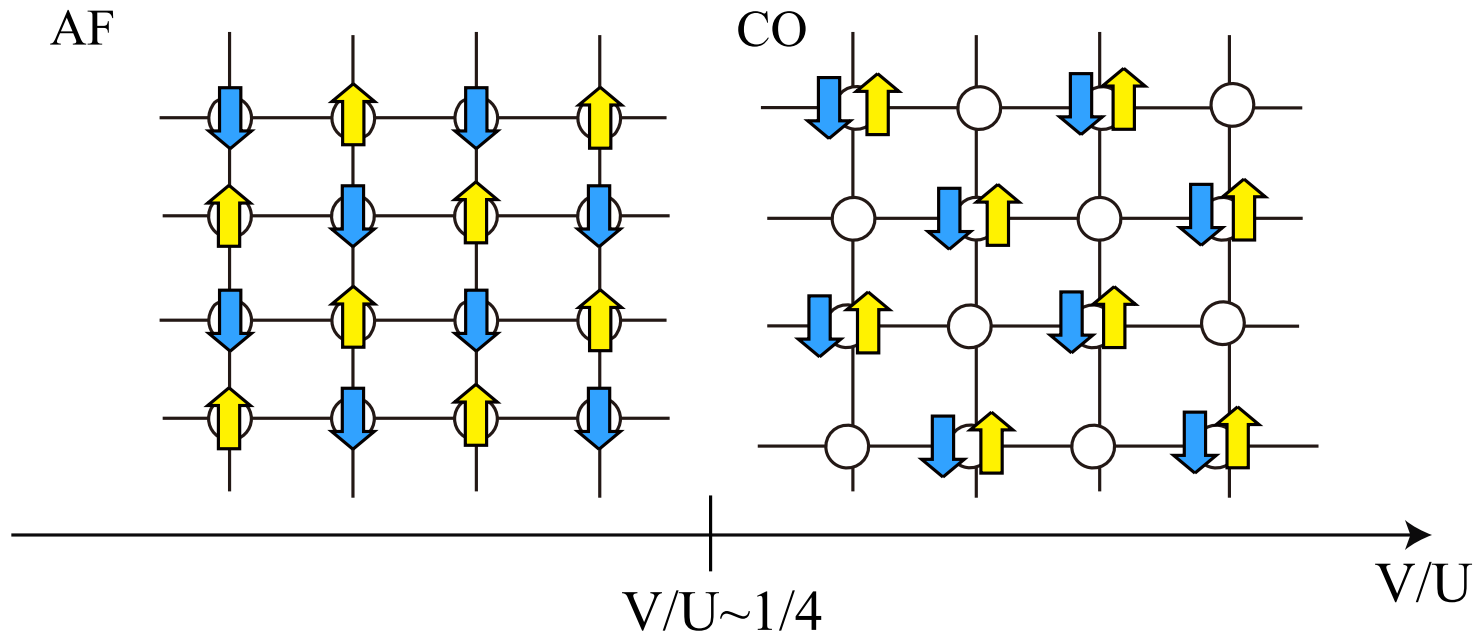
発展 [様々な量子相]

1. Hubbard + $V \rightarrow$ 電荷秩序
2. Heisenberg+ $J_2 \rightarrow$ ストライプ磁気秩序
3. Attractive Hubbard \rightarrow 1s 超伝導
4. 1D Kondo lattice \rightarrow 近藤絶縁体
5. Kitaev model \rightarrow Kitaev spin 液体

4, 5は少し計算が重いです.

1. オフサイト斥力 V の導入

$$H_V = V \sum_{\langle i,j \rangle} n_i n_j$$



電荷秩序が起きるかどうかをResult_Nq.dat
もしくはfourier tool を使って確認しましょう

1. オフサイト斥力 V の導入

$$H_V = V \sum_{\langle i,j \rangle} n_i n_j$$

StdFace.def の例

W	=	4
L	=	4
Wsub	=	2
Lsub	=	2
model	=	"FermionHubbard"
lattice	=	"Tetragonal"
t	=	1.0
U	=	4.0
V	=	2.0
nelec	=	16
2Sz	=	0
NVMCSample	=	50
NSROptItrStep	=	500
NSROptItrSmp	=	50
NMPTrans	=	4

書き換え方の例

1. `perl -w MakeMod.pl`
でStdFace.defを生成

2. 生成したStdFace.def,
StdFace_2.def, StdFace_aft.def
でV= 2.0 を追加

3. X.shの中の

`perl -w MakeMod.pl`
をコメントアウト

`#perl -w MakeMod.pl`

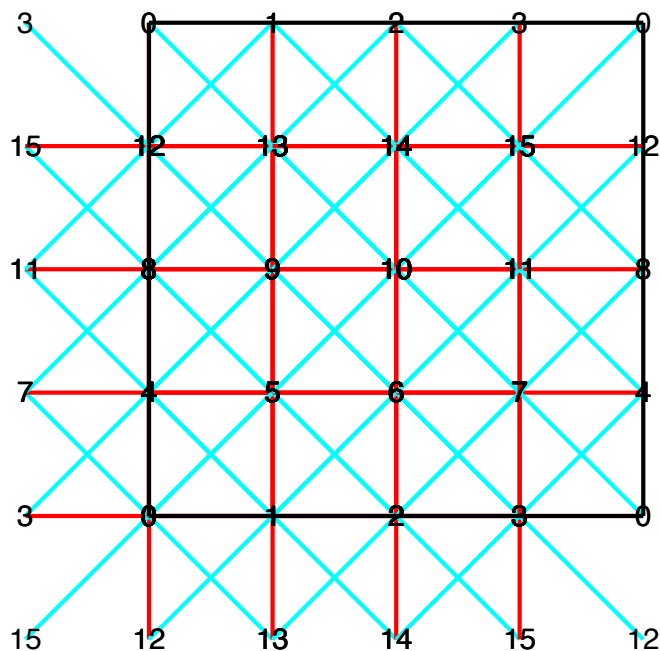
4. `sh ./X.sh`

もしくはMakeMod.plを
直接書き換える

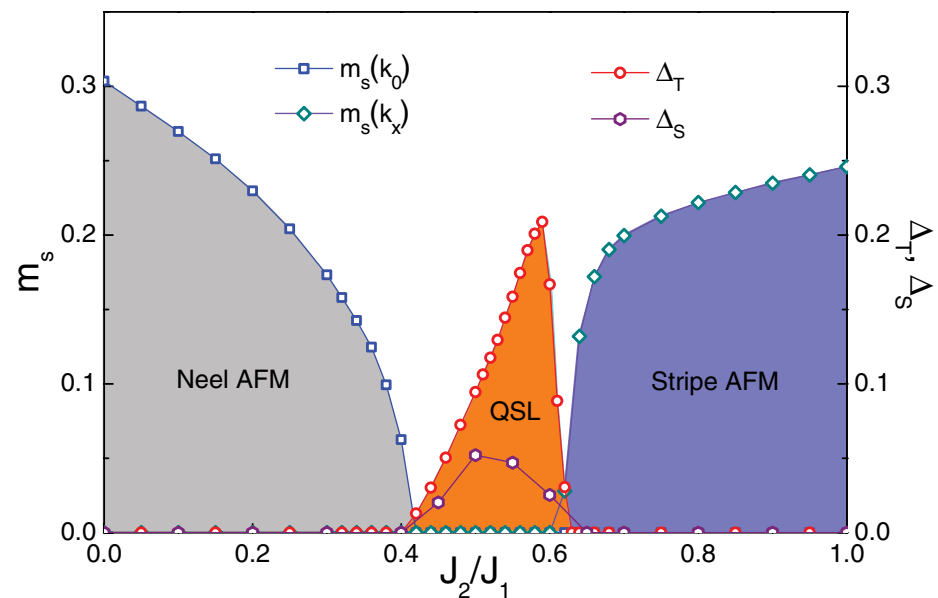
2. J1-J2ハイゼンベルク模型

$$H = J_1 \sum_{\langle i,j \rangle} S_i S_j + J_2 \sum_{\langle\langle i,j \rangle\rangle} S_i S_j$$

最近接 J1, 次近接 J2



lattice.gpで描画可能



PRB 86, 024424(2012)

J2/J1~0.5で非磁性の
基底状態(スピン液体?)

2. J1-J2ハイゼンベルク模型

- J2=1でストライプ磁気秩序が起きるかどうかを

Result_Sq.datもしくはfourier tool を使って確認しましょう

- J2=0.5近傍だとどうなるでしょうか？

StdFace.def の例

```
W          = 4
L          = 4
Wsub       = 2
Lsub       = 2
model      = "Spin"
lattice     = "Tetragonal"
J          = 1.0
J'         = 1.0
2Sz        = 0
NVMCSample = 50
NSROptItrStep = 500
NSROptItrSmp = 100
NMPTrans   = 1
NSPGaussLeg = 1
```

書き換え方の例

1. `perl -w MakeMod.pl`
でStdFace.defを生成

2. 生成したStdFace.def,
StdFace_2.def, StdFace_aft.def
でJ'= 1.0 を追加

3. X.shの中の

`perl -w MakeMod.pl`
をコメントアウト

`#perl -w MakeMod.pl`

4. `sh ./X.sh`

3.attractive Hubbard model

- Uを負にするだけ
- 1s (等方的)の超伝導相関を計算してみましょう

(half fillingだと電荷秩序と超伝導が共存しているので, dopeした方がよいです)

書き換え方の例

1. perl -w MakeMod.pl
でStFace.defを生成

2. 生成したStdFace.def,
StdFace_2.def, StdFace_aft.def
でU=-4.0 に変更

3. X.shの中の

perl -w MakeMod.pl
をコメントアウト

#perl -w MakeMod.pl

4. sh ./X.sh

$$P_{1s}(\vec{r}) = \frac{1}{2N_s} \sum_{i=1}^{N_s} \langle \Delta_{1s}^\dagger(\vec{r}_i) \Delta_{1s}(\vec{r}_i + \vec{r}) + \Delta_{1s}(\vec{r}_i) \Delta_{1s}^\dagger(\vec{r}_i + \vec{r}) \rangle$$

$$\Delta_{1s}(\vec{r}_i) = \frac{1}{\sqrt{2}} (c_{i\uparrow} c_{i\downarrow} - c_{i\downarrow} c_{i\uparrow}).$$

- 1sの相関関数を計算する
スクリプトは以下にあります
Samples/SC_Correlation

使い方

1. sh X.sh

2. perl -w ForSCCor.pl → SC_cisajscktaltdc.def

3. namelist.defの

TwoBodyG greentwo.def → TwoBodyG SC_cisajscktaltdc.def

4. vmc.out namelist.def

5. ln -s output SC

6. perl -w Aft_SC1.pl

7. plot "Ave_Max_SC_Sum_1s_0.dat" w e

4. 1D Kondo lattice model

StdFace.def の例

```
L = 8
Lsub = 2
model = "Kondo"
lattice = "chain"
t = 1.0
J = 1
nelec = 8
2Sz = 0
NVMCSample = 200
NSROptItrStep = 2000
NSROptItrSmp = 200
NMPTtrans = 2
NSPGaussLeg = 8
```

$S=0$, $S=1$ の計算をして

スピンギャップを計算してみましょう

$(S = 0)$	E/N_s	S_{loc}	S_{nn}^f	$S(\pi)$
ED	-1.394104	-0.3151569745	-0.3386218911	0.05685112698
mVMC(2)	-1.39352(1)	-0.3142(6)	-0.3365(2)	0.0575(1)
mVMC(2)+Lanczos	-1.39399(2)	-	-	-
mVMC(8)	-1.39400(1)	-0.3158(3)	-0.3382(3)	0.0567(1)
mVMC(8)+Lanczos	-1.394099(5)	-	-	-
$(S = 1)$	E/N_s	S_{loc}	S_{nn}^f	$S(\pi)$
ED	-1.382061	-0.27480917	-0.224015671	0.057478
mVMC(2)	-1.38128(3)	-0.2743(3)	-0.2248(4)	0.05811(6)
mVMC(2)+Lanczos	-1.38187(1)	-	-	-
mVMC(8)	-1.38171(3)	-0.2750(4)	-0.2249(7)	0.0577(1)
mVMC(8)+Lanczos	-1.382011(2)	-	-	-

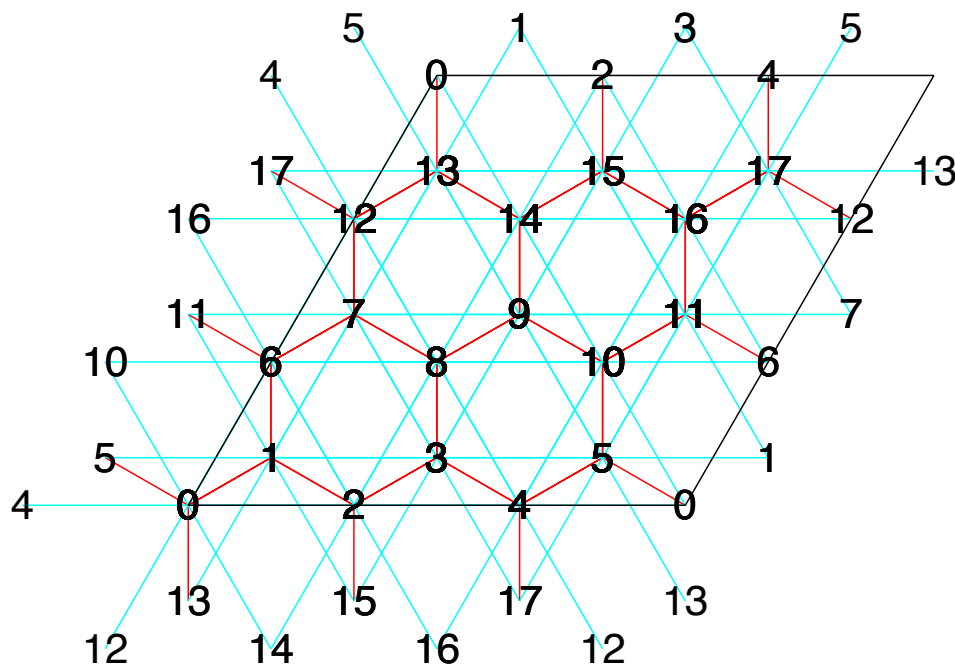
Table 5: Comparisons with exact diagonalization for one-dimensional Kondo-lattice model for $J = 1$ and $t = 1$. Upper (Lower) panel shows the results for spin singlet (triplet) sector. In the triplet sector ($S = 1$), we take total momentum as $K = \pi$, which gives the lowest energy in $S = 1$, while we take total momentum as $K = 0$ for $S = 1$.

See, H. Tsunetsugu *et al.*, PRB 46, 3175 (1992),
H. Tsunetsugu *et al.*, RMP 69, 809 (1997)

5: Kitaev model

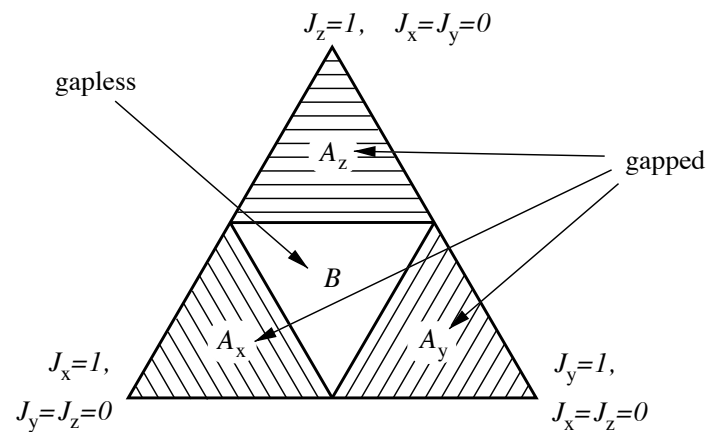
$$H = -J_x \sum_{x\text{-bond}} S_i^x S_j^x - J_y \sum_{y\text{-bond}} S_i^y S_j^y - J_z \sum_{z\text{-bond}} S_i^z S_j^z$$

3方向のそれぞれが
 J_x, J_y, J_z で相互作用



lattice.jpで描画可能

相図



Annals of Physics 321, 2-111 (2016)

可解模型→スピン液体

5: Kitaev model

$$H = -J_x \sum_{x\text{-bond}} S_i^x S_j^x - J_y \sum_{y\text{-bond}} S_i^y S_j^y - J_z \sum_{z\text{-bond}} S_i^z S_j^z$$

StdFace.def の例

```
W          = 2
L          = 3
model      = "SpinGC"
lattice    = "Honeycomb"
NVMCSample = 200
NSROptItrStep = 5000
NSROptItrSmp = 100
NMPTTrans  = 6
J0x        = -1.0
J0y        = 0.0
J0z        = 0.0
J1x        = 0.0
J1y        = -1.0
J1z        = 0.0
J2x        = 0.0
J2y        = 0.0
J2z        = -1.0
```

乱数スタートは収束が遅いので注意

→収束をどうしたら加速するか？

[open problem]