Introduction to HΦ –A numerical solver for quantum lattice models

三澤　貴宏

東京大学物性研究所計算物質科学研究センター

計算物質科学人材育成コンソーシアム(PCoMS) PI

Outline

1. What can we do by HΦ ?
2. [How to get HΦ]
3. How to use Standard mode
4. How to use Expert mode
5. Applications of HΦ
6. [Short introduction to mVMC]

http://ma.cms-initiative.jp/ja/listapps/hphi

Developers of HΦ

M. Kawamura T. Misawa K. Yoshimi

Y. Yamaji

S. Todo

N. Kawashima

Development of HΦ is supported by "*Project for advancement of software usability in materials science*" by ISSP

What can we do by HΦ?

For Hubbard model, spin-*S* Heisenberg model, Kondo-lattice model *with arbitrary one-body and two-body interactions*

- Full diagonalization

- Ground state calculations by Lanczos method

- Finite-temperature calculations by thermal pure quantum (TPQ) states

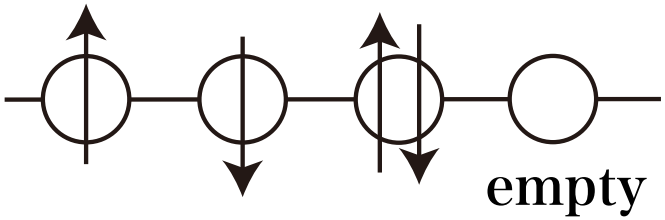
- Dynamical properties (optical conductivity ..)

maximum system sizes@ ISSP system B (sekirei)

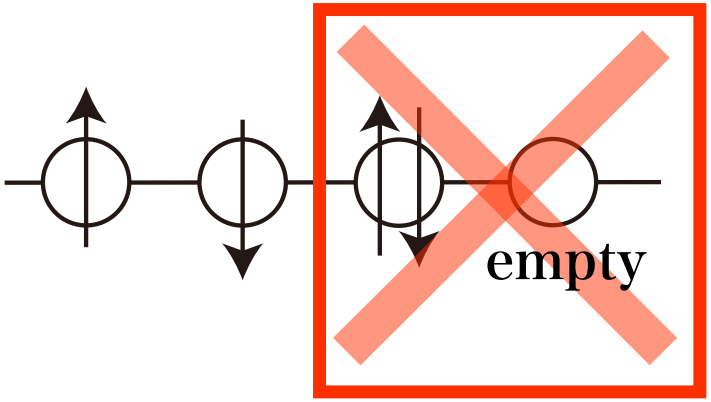
- spin 1/2: ~ 40 sites (Sz conserved)
- Hubbard model: ~ 20sites (# of particles & Sz conserved)

Available models in HΦ

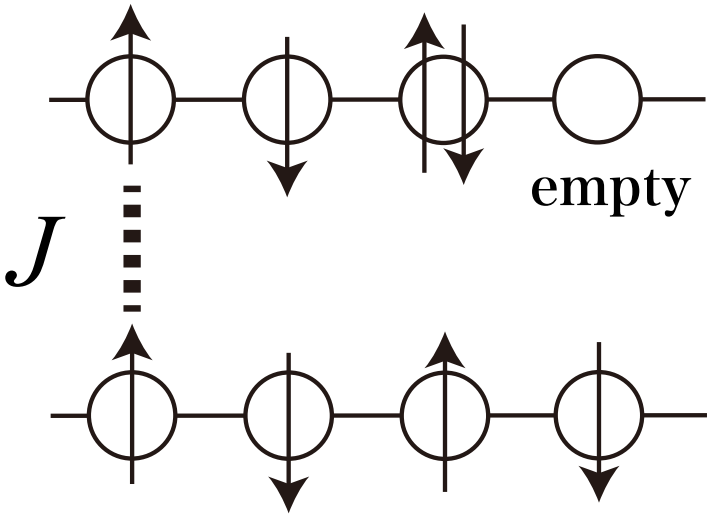
Hubbard (itinerant) ~ 4^N



Heisenberg (localized) ~ 2^N



Kondo=itinerant+localized



３つの異なる模型を扱えるように整備

(Heisenbergはspin-Sも対応)

Descriptions of quantum models

e.g. Hubbard model

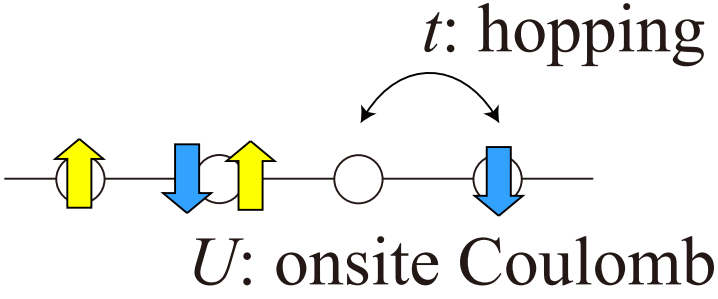
$$\hat{\mathcal{H}} = \hat{\mathcal{H}}_t + \hat{\mathcal{H}}_U$$

Electrons as waves

$$\hat{H}_t = -t \sum_{\langle i,j \rangle, \sigma} (\hat{c}^\dagger_{i\sigma} \hat{c}_{j\sigma} + \hat{c}^\dagger_{j\sigma} \hat{c}_{i\sigma})$$

Electrons as particles

$$\hat{H}_U = \sum_i \hat{n}_{i\uparrow} \hat{n}_{i\downarrow}, \;\; \hat{n}_{i\sigma} = \hat{c}^\dagger_{i\sigma} \hat{c}_{i\sigma}$$



Relations between 2nd-quantized operators (these are all !)

$$\{\hat{c}^\dagger_{i\sigma}, \hat{c}_{j\sigma'}\} = \hat{c}^\dagger_{i\sigma} \hat{c}_{j\sigma'} + \hat{c}_{j\sigma'} \hat{c}^\dagger_{i\sigma} = \delta_{i,j} \delta_{\sigma,\sigma'}$$

$$\{\hat{c}^\dagger_{i\sigma}, \hat{c}^\dagger_{j\sigma'}\} = 0 \rightarrow \hat{c}^\dagger_{i\sigma} \hat{c}^\dagger_{i\sigma} = 0$$

Pauli's principle

$$\{\hat{c}_{i\sigma}, \hat{c}_{j\sigma'}\} = 0 \rightarrow \hat{c}_{i\sigma} \hat{c}_{i\sigma} = 0$$

Full diagonalization by *hand*

Matrix representation of Hamiltonian (ex. 2 site Hubbard model)

Real-space configuration $|\uparrow,\downarrow\rangle = c^{\dagger}_{1\uparrow}c^{\dagger}_{2\uparrow}|0\rangle$

After some *tedious* calculations, $\langle\uparrow,\downarrow|\hat{H}_t|\uparrow\downarrow,0\rangle = \langle\uparrow,\downarrow|(t\sum_{\sigma} c^{\dagger}_{1\sigma}c_{2\sigma} + c^{\dagger}_{2\sigma}c_{1\sigma})|\uparrow\downarrow,0\rangle = -t$

$$\mathcal{H} = \begin{array}{c} \\ \langle\uparrow,\downarrow| \\ \langle\downarrow,\uparrow| \\ \langle\uparrow\downarrow,0| \\ \langle 0,\uparrow\downarrow| \end{array} \begin{array}{c} \begin{array}{cccc} |\uparrow,\downarrow\rangle & |\downarrow,\uparrow\rangle & |\uparrow\downarrow,0\rangle & |0,\uparrow\downarrow\rangle \end{array} \\ \begin{pmatrix} 0 & 0 & -t & -t \\ 0 & 0 & t & t \\ -t & t & U & 0 \\ -t & t & 0 & U \end{pmatrix} \end{array}$$

$$|\phi\rangle = a_0|\uparrow,\downarrow\rangle + a_1|\downarrow,\uparrow\rangle + a_2|\uparrow\downarrow,0\rangle + a_3|0,\uparrow\downarrow\rangle$$

Diagonalization → eigenvalues, eigenvectors
→ Problem is completely solved (НФ)

Full diagonalization by HΦ

Matrix representation of Hamiltonian

(real space basis) → Full diagonalization for the matrix

$$H_{ij} = \langle i|\hat{H}|j\rangle \qquad |i\rangle \text{ real-space basis}$$

HΦ automatically generates matrix elements !

[2-digit binary number & bit operations]

dim. of matrix= # of real-space bases

=exponentially large

ex. spin1/2 system: $S_z=0$ $\quad {}_{N_s}C_{N_s/2}$

- Ns=16: dim.=12800, required memory (~dim.^{2}) ~ 1 GB
- Ns=32: dim.~6×10^8 , required memory (~dim.^{2}) ~ 3 EB!

Lanczos method

By multiplying the Hamiltonian to initial vector, we can obtain the ground state (power method)

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

A few (at least two) vectors are necessary→

We can treat larger system size than full diagonalization !

ex. spin 1/2 system: S_z=0

- Ns=16: dim. =12800,required memory (~dim.) ~0.1 MB
- Ns=32: dim. ~6×10^8,required memory (~dim.) ~5 GB !
- Ns=36: dim. ~9×10^9,required memory (~dim.) ~72 GB !

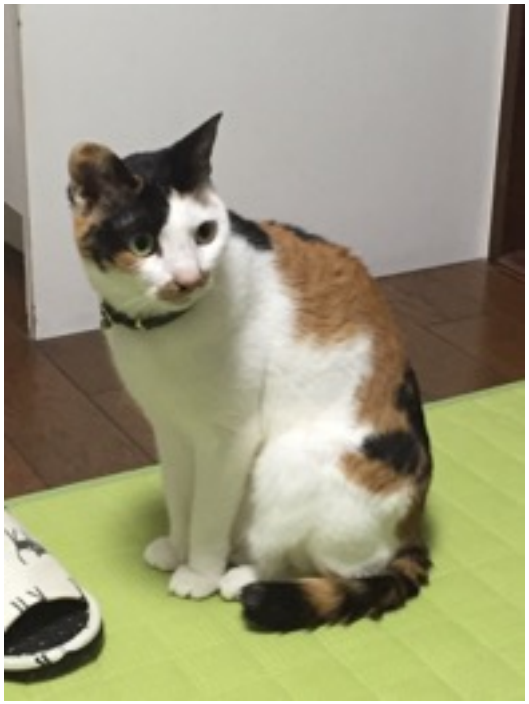
Meaning of name & logo

- Multiplying H to Φ (HΦ)
- This cat means wave function in two ways

cat is a symbol of superposition.. (Schrödinger's cat)

waking **sleeping**

Φ=



+

Finite-temperature calculations by TPQ

-Conventional finite-temperature cal.:

ensemble average is necessary

→ Full diag. is necessary

$$\langle E \rangle = \frac{\sum_n E_n e^{-\beta E_n}}{\sum_n e^{-\beta E_n}}$$

It is shown that thermal pure quantum state (TPQ) states enable us to calculate the physical properties at finite temperatures w/o ensemble average [Sugiura-Shimizu, PRL 2012,2013]

→ Cost of finite-tempeature calculations ~ Lanczos method !

pioneering works :
Quantum-transfer MC method (Imada-Takahashi, 1986),
Finite-temperature Lanczos (Jaklic-Prelovsek,1994),
Hams-Raedt (2000)

Sugiura-Shimizu method [mTPQ state]

Procedure

S. Sugiura and A. Shimizu, PRL 2012 & 2013

$|\psi_0\rangle$: random vector

$$|\psi_k\rangle \equiv \frac{(l - \hat{H}/N_{\rm s})|\psi_{k-1}\rangle}{|(l - \hat{H}/N_{\rm s})|\psi_{k-1}\rangle|}$$

***l*:constant larger the maximum eigenvalues**

$$u_k \sim \langle\psi_k|\hat{H}|\psi_k\rangle/N_{\rm s}$$

$$\beta_k \sim \frac{2k/N_{\rm s}}{(l - u_k)}, \quad \langle\hat{A}\rangle_{\beta_k} \sim \langle\psi_k|\hat{A}|\psi_k\rangle$$

All the finite temperature properties can be calculated by using *one* thermal pure quantum [TPQ] state.

Essence of TPQ

cf. 二重ヒルベルト空間 (熱場ダイナミクス)
鈴木増雄, 統計力学(岩波書店); 高橋康, 物性研究 20, 97(1973)

1. Random vector (high-temperature limit) *equally* includes all eigenvectors

$$|\Phi_{\rm rand}\rangle = \sum_n a_n |n\rangle, |a_n|^2 \sim 1/N_{\rm H}$$

$$\langle \hat{H} \rangle = \frac{\langle \Phi_{\rm rand}|\hat{H}|\Phi_{\rm rand}\rangle}{\langle \Phi_{\rm rand}|\Phi_{\rm rand}\rangle} = \frac{\sum_n |a_n|^2 E_n}{\sum_n |a_n|^2} \sim \frac{1}{N_{\rm H}} \sum_n E_n$$

2. Commutative quantities can be calculated by *single wave function*

$$|\Phi(\beta)\rangle = e^{-\beta \hat{H}/2} |\Phi_{\rm rand}\rangle$$

$$\frac{\langle \Phi(\beta)|\hat{H}|\Phi(\beta)\rangle}{\langle \Phi(\beta)|\Phi(\beta)\rangle} = \frac{\sum_n |a_n|^2 E_n e^{-\beta E_n}}{\sum_n |a_n|^2 e^{-\beta E_n}} \sim \frac{\sum_n E_n e^{-\beta E_n}}{\sum_n e^{-\beta E_n}} = E(\beta)$$

3. Non-commutative quantities can be also calculated by *single wave function*

Proofs: Hams and De Raedt PRE 2000; Sugiura and Shimizu PRL 2012,2013

Thermal Pure Quantum state (熱的純粋量子状態) by Sugiura and Shimizu

Drastic reduction of numerical cost

Heisenberg model, 32 sites, S_z=0

Full diagonalization:
Dimension of Hamiltonian ~ $10^8 \times 10^8$
Memory ~ 3E Byte → Almost impossible.

TPQ method:
Only two vectors are required:
dimension of vector ~ 10^8 ~~$\times 10^8$~~
Memory ~ 10 G Byte
→ Possible even in lab's cluster machine !

Basic properties of HΦ

What can we do by HΦ?

For Hubbard model, spin-*S* Heisenberg model, Kondo-lattice model

- Full diagonalization

- Ground state calculations by Lanczos method

- Finite-temperature calculations by thermal pure quantum (TPQ) states

- Dynamical properties (optical conductivity ..)

maximum system sizes@ ISSP system B (sekirei)

- spin 1/2: ~ 40 sites (Sz conserved)
- Hubbard model: ~ 20sites (# of particles & Sz conserved)

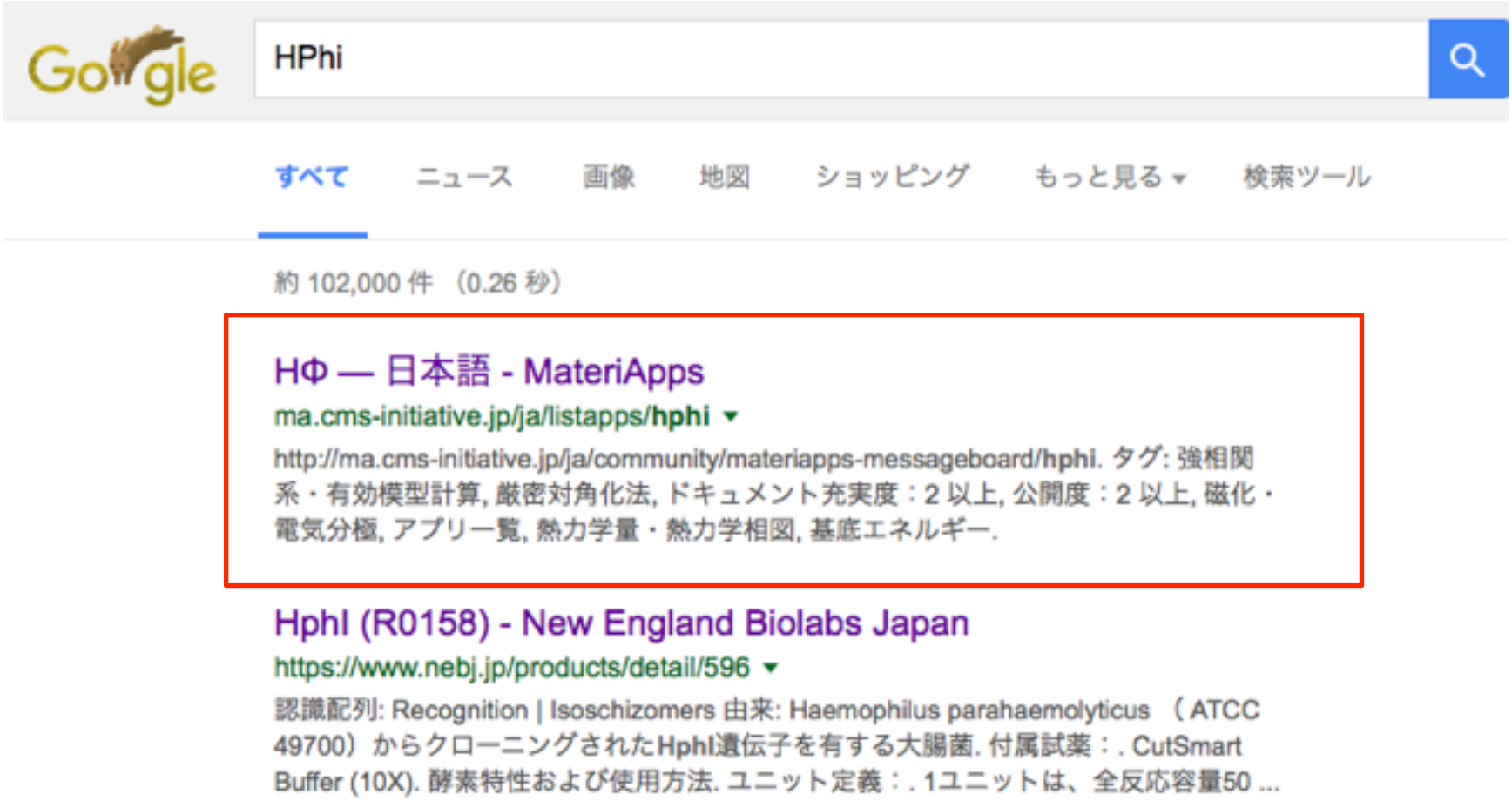

Let’s get НФ !
НФ

How to find HΦ

http://ma.cms-initiative.jp/en/application-list/hphi/hphi

search by “HPhi” → You can find our homepage in the first page (maybe, the first or second candidate)

GitHub → https://github.com/QLMS/HPhi



How to compile HΦ

ex. linux + gcc-mac

```
tar xzvf HPhi-release-1.2.tar.gz
cd HPhi-release-1.2
bash HPhiconfig.sh gcc-mac
make HPhi
```

For details,

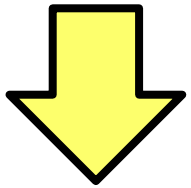
```
$ bash HPhiconfig.sh

Usage:
./HPhiconfig.sh system_name
 system_name should be chosen from below:
     sekirei : ISSP system-B
        maki : ISSP system-C
       intel : Intel compiler + Linux PC
 mpicc-intel : Intel compiler + Linux PC + mpicc
         gcc : GCC + Linux
     gcc-mac : GCC + Mac
```

Let's start HΦ !
(Standard mode)

How to use HΦ: Standard mode I (Lanczos)

Only StdFace.def is necessary (< 10 lines) !

```
L          = 4
model      = "Spin"
method     = "Lanczos"
lattice    = "square lattice"
J          = 1.0
2Sz        = 0
```



HPhi -s StdFace.def

ex. 4×4 2d Heisenberg model,
GS by Lanczos method

Method

Lanczos — ground state

TPQ — finite-temperature

FullDiag — full-diagonalization

./ouput : results are output

Important files

./output/zvo_energy.dat → energy

./output/zvo_Lanczos_Step.dat → convergence

./output/zvo_cisajs.dat → one-body Green func.

./output/zvo_cisajscktalt.dat → two-body Green func.

How to use HΦ: Standard mode II

./output/zvo_energy.dat

ex. 4by4, 2d Heisenberg model,
GS calculations by Lanczos

```
$ cat output/zvo_energy.dat
Energy  -11.2284832084288109
Doublon   0.0000000000000000
Sz        0.0000000000000000
```

GS energy

./output/zvo_Lanczos_Step.dat

```
$ tail output/zvo_Lanczos_Step.dat
stp=28 -11.2284832084 -9.5176841765 -8.7981539671 -8.5328120558
stp=30 -11.2284832084 -9.5176875029 -8.8254961060 -8.7872255591
stp=32 -11.2284832084 -9.5176879460 -8.8776934418 -8.7939798590
stp=34 -11.2284832084 -9.5176879812 -8.8852955092 -8.7943260103
stp=36 -11.2284832084 -9.5176879838 -8.8863380562 -8.7943736678
stp=38 -11.2284832084 -9.5176879839 -8.8864307327 -8.7943782609
stp=40 -11.2284832084 -9.5176879839 -8.8864405361 -8.7943787937
stp=42 -11.2284832084 -9.5176879839 -8.8864422628 -8.7943788984
stp=44 -11.2284832084 -9.5176879839 -8.8864424018 -8.7943789077
stp=46 -11.2284832084 -9.5176879839 -8.8864424075 -8.7943789081
```

convergence process by Lanczos method

How to use HΦ: Standard mode III

./output/zvo_cisajs.dat

$\langle c^\dagger_{i\sigma} c_{j\tau} \rangle$

```
$ head output/zvo_cisajs.dat
    0    0    0    0 0.5000000000 0.0000000000
    0    1    0    1 0.5000000000 0.0000000000
```

$\langle c^\dagger_{0\downarrow} c_{0\downarrow} \rangle$

$\langle c^\dagger_{0\uparrow} c_{0\uparrow} \rangle$

./output/zvo_cisajscktalt.dat

```
$ head output/zvo_cisajscktalt.dat
    0    0    0    0    0    0    0    0 0.5000000000 0.0000000000
    0    0    0    0    0    1    0    1 0.0000000000 0.0000000000
    0    0    0    0    1    0    1    0 0.1330366332 0.0000000000
    0    0    0    0    1    1    1    1 0.3669633668 0.0000000000
```

$\langle c^\dagger_{0\downarrow} c_{0\downarrow} c^\dagger_{0\downarrow} c_{0\downarrow} \rangle$

$\langle c^\dagger_{0\downarrow} c_{0\downarrow} c^\dagger_{0\uparrow} c_{0\uparrow} \rangle$

$\langle c^\dagger_{0\downarrow} c_{0\downarrow} c^\dagger_{1\downarrow} c_{1\downarrow} \rangle$

$\langle c^\dagger_{0\downarrow} c_{0\downarrow} c^\dagger_{1\uparrow} c_{1\uparrow} \rangle$

ex. onsite・nn-site correlation func.

How to use HΦ: Standard mode IV

HPhi/samples/Standard/
StdFace.def for
Hubbard model, Heisenberg model, Kitaev model,
Kondo-lattice model

By changing StdFace.def slightly, you can easily perform the calculations for different models.

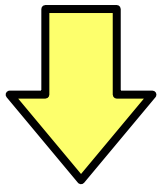
Cautions：

- Do not input too large system size
(upper limit@laptop: spin 1/2→24 sites, Hubbard model 12 sites)
- Lanczos method is unstable for too small size
(dim. > 1000)
-TPQ method does not work well for small size
(dim. > 1000)

Expert mode !

How to use HΦ: What is Expert mode ?

HPhi -s StdFace.def



Standard mode: Necessary input files are automatically generated

Files for Hamiltonian (three files)
zInterAll.def,zTrans.def, zlocspn.def

Files for basic parameters (two files)
modpara.def,calcmod.def

Files for correlations functions (two files)
greenone.def, greentwo.def

+ list of input files: namelist.def

Expert mode: preparing the following files by yourself

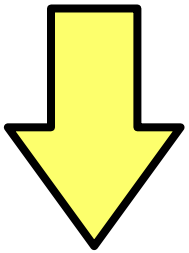
How to use HΦ: What is Expert mode ?

Expert mode: preparing the following files by yourself

Files for Hamiltonian (three files)
zInterAll.def,zTrans.def, zlocspn.def

Files for basic parameters (two files)
modpara.def,calcmod.def

Files for correlations functions (two files)
greenone.def, greentwo.def



execute following command

HPhi -e namelist.def

How to use HΦ: zInterall.def

Examples of input files for Hamiltonian

$$H+=\sum_{i,j,k,l}\sum_{\sigma_1,\sigma_2,\sigma_3,\sigma_4} I_{ijkl\sigma_1\sigma_2\sigma_3\sigma_4} c^{\dagger}_{i\sigma_1} c_{j\sigma_2} c^{\dagger}_{k\sigma_3} c_{l\sigma_4}$$

```
======================
NInterAll      96
======================
========zInterAll=====
======================
```

# of interactions

								real	imaginary
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
...									
i	σ_1	j	σ_2	k	σ_3	l	σ_4		

You can specify *arbitrary* two-body interactions
→ You can treat *any* lattice structures

How to use HΦ: Expert mode

Simple version of zInterall.def

- CoulombIntra $H+=\sum_i U_i n_{i\uparrow} n_{i\downarrow}$

```
================================
NCoulombintra 2
================================
==========Exchange=============
================================
 0   4.0
 1   4.0
```

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

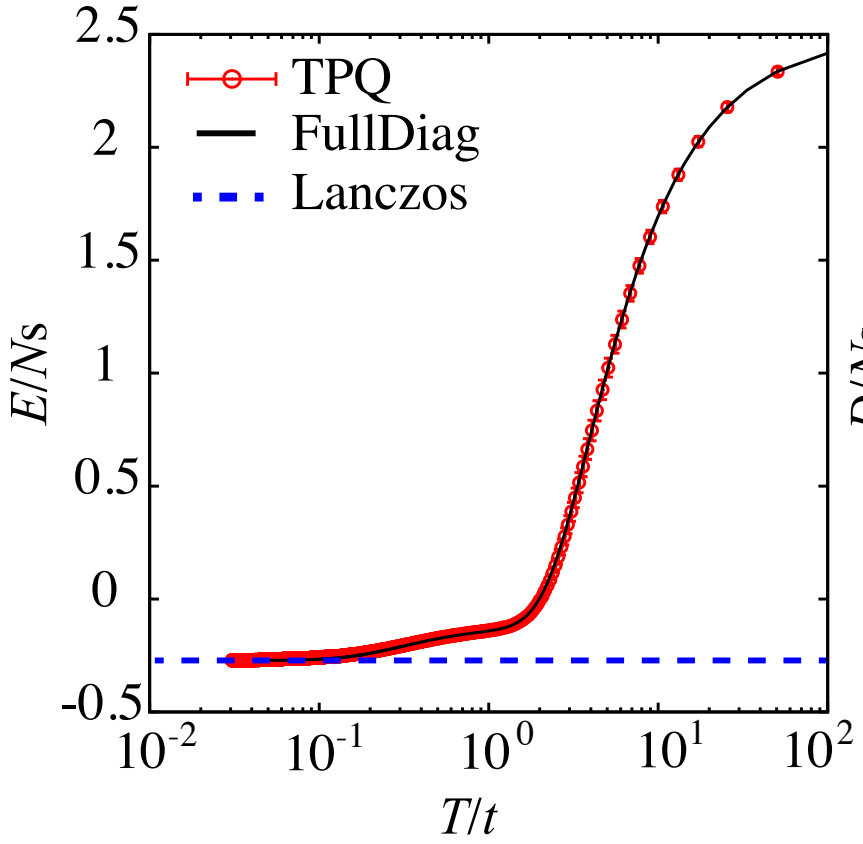
```
================================
NExchange 2
================================
==========Exchange=============
================================
 0   1  0.5
 1   2  0.5
```

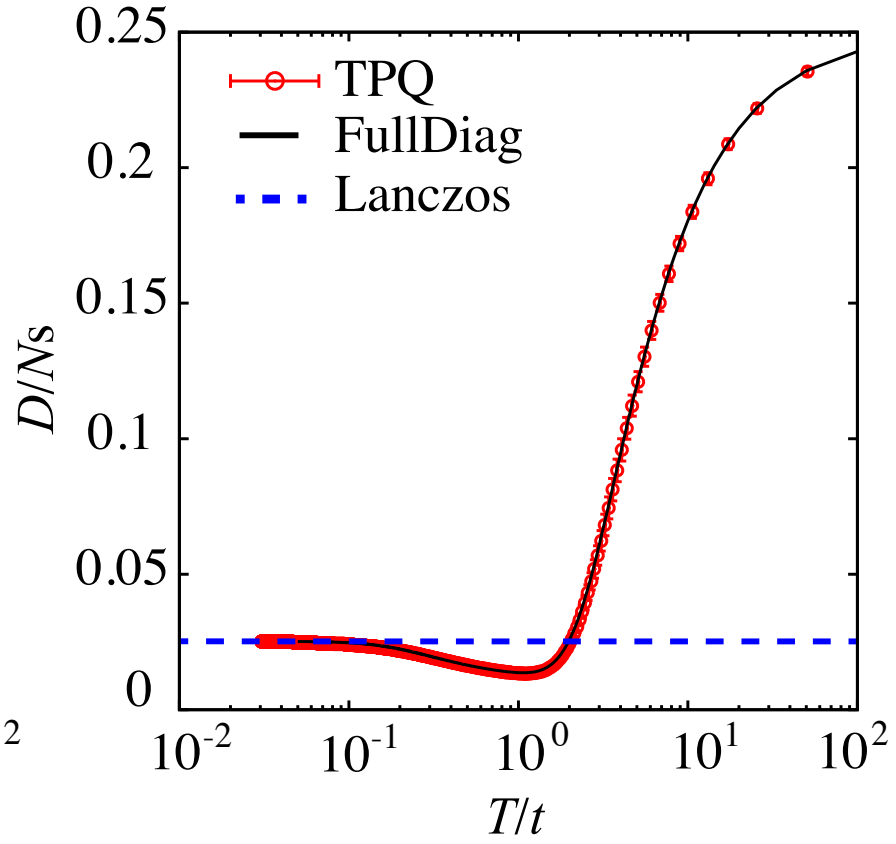
Easy to input interactions


Applications of НФ!

Comparison of three different methods

Comparison of FullDiag, TPQ, Lanczos method
Hubbard model, L=8, U/t=8, half filling, S_z=0





TPQ method works well !

Studies using HPhi

1. Finite-temperature crossover phenomenon in the S=1/2 antiferromagnetic Heisenberg model on the kagome lattice
Tokuro Shimokawa, Hikaru Kawamura: J. Phys. Soc. Jpn. 85, 113702 (2016)

2. Finite-Temperature Signatures of Spin Liquids in Frustrated Hubbard Model
Takahiro Misawa, Youhei Yamaji (arXiv:1608.09006)

3. Four-body correlation embedded in antisymmetrized geminal power wave function
Airi Kawasaki, Osamu Sugino, The Journal of Chemical Physics 145, 244110 (2016)

4. Liquid-Liquid Transition in Kitaev Magnets Driven by Spin Fractionalization
Joji Nasu, Yasuyuki Kato, Junki Yoshitake, Yoshitomo Kamiya, Yukitoshi Motome, Phys. Rev. Lett. 118,137203 (2017)

既に、4本の論文がHPhiを使用！

HPhiの使い方

0. 汎用性を優先して、速度・サイズなどは犠牲にしている部分がある→
対角化(Lanczos法)での世界最大の計算は（現段階では)無理

1. spin 1/2 36 sites, Hubbard 18 sites程度までの有限・励起状態計算は比較的すぐできる。とくに、エントロピーが低温まで残るフラストレート系が得意 [論文 1(kagome),2(*t*-*t*' Hubbard)]。

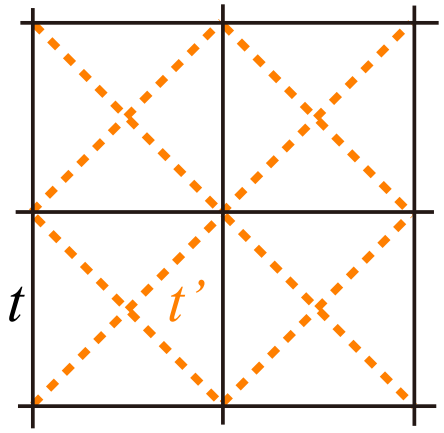
2. 平均場計算などで「面白い」ことがおきることを確認
→HPhiでその結果を確認する[論文4(extended Kitaev model)]

3. 新手法開発した際の精度確認[論文3(extended geminal wave functions)]
~20 site Hubbard model

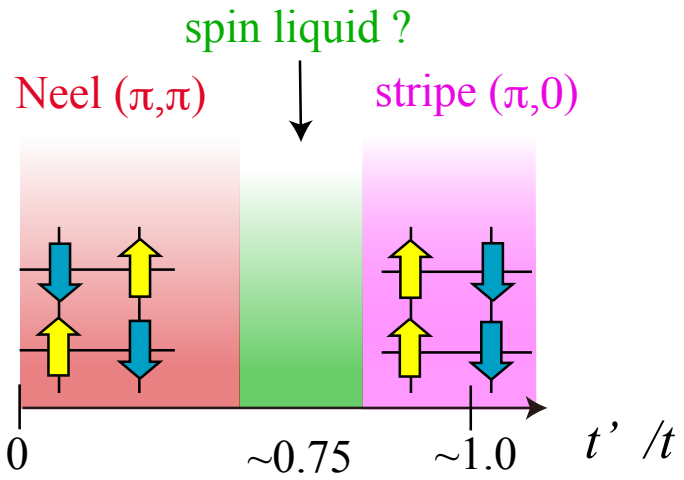
4. 新奇物質に対する現実的な有効模型の妥当性の確認,物性予測
(励起状態、有限温度、動的物理量)[Na_2IrO_3, Yamaji *et al.*]

Frustrated *t-t'* Hubbard model

Lattice geometry

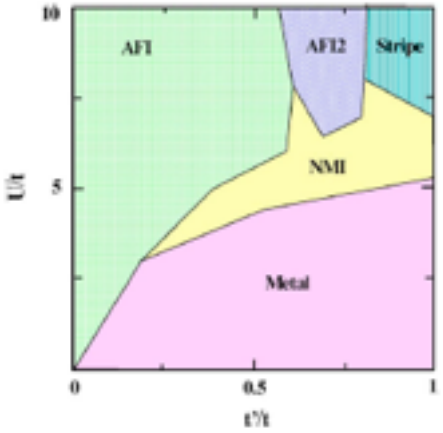


Schematic phase diagram

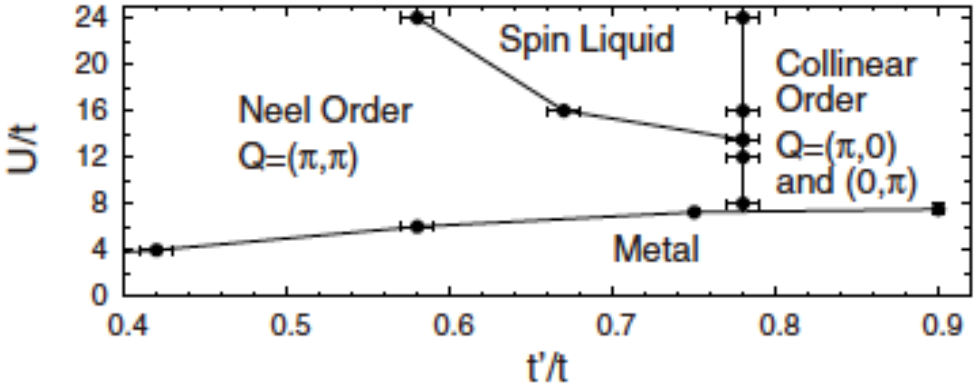


Previous studies

PIRG: Mizusaki and Imada, PRB 2004



VMC: L. Tocchio *et al.*, PRB(R) 2008



NB: Spin liquid is also reported in J_1-J_2 Heisenberg model

Spin liquid may appear at intermediate region

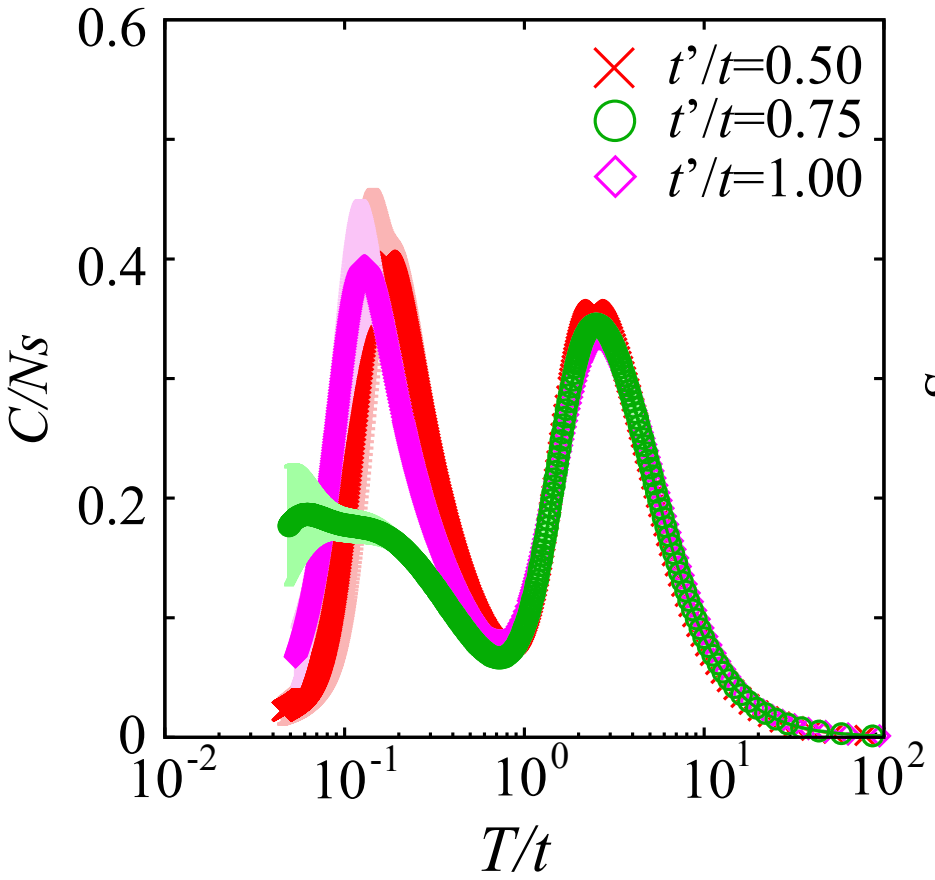
Input file

```
W = 4
L = 4
model = "FermionHubbard"
method = "TPQ"
lattice = "Tetragonal"
t  = 1.0
t' = 0.75
U  = 10.0
nelec = 16
2Sz = 0
```

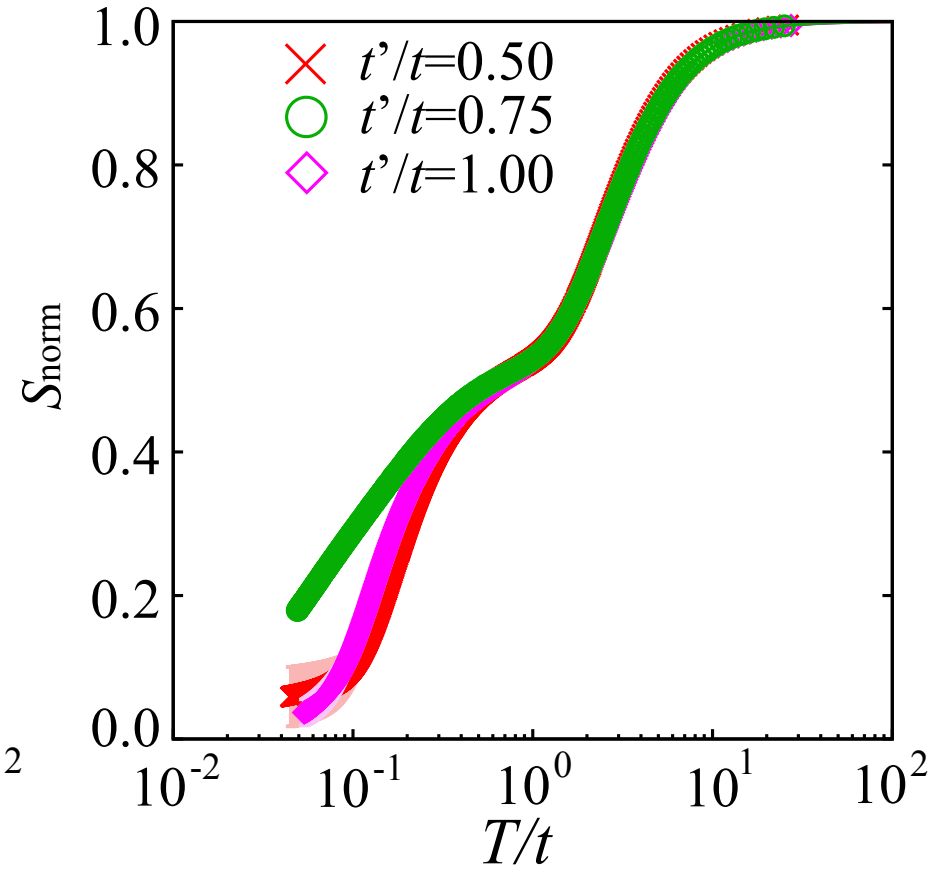
たった、これだけ！そのまま並列計算も可能

Signature of spin liquid [U/t=10]

specific heat



entropy



At t'/t~0.75 large entropy remains at low temperatures
→ Signature of spin liquid

Available system size in SC@ISSP

ISSP system B (sekirei)

✓fat node: 1node (40 cores) memory/node = 1TB、
up to 2nodes → ~2TB

✓cpu node: 1node (24cores) memory/node=120GB,
up to 144nodes→~17TB

SC@ISSP:

- It is *very easy (cheap)* to perform the calculations up to spin 1/2 = 32 sites, Hubbard = 16 sites

- It is possible (but expensive !) to perform the calculations up to spin 1/2 40 sites, Hubbard 20 sites
(state-of-the-art calculations 5-10 years ago)

Summary

- Explained basic properties of HΦ:
Full diagonalization, Lanzcos method, TPQ method
for Heisenberg, Hubbard, Kondo, Kitaev model ….

- Explained how to use HΦ:
Very easy to start calculations by using Standard mode
Easy to treat general Hamiltonians by using Expert mode

- Shown applications of HΦ:
Found the finite-temperature signature of QSL in
t-*t*' Hubbard model

If you have any questions,
please join HPhi ML and ask questions