Selected Topics in Computational Quantum Physics

量子物理计算方法选讲

Shuo Yang（杨硕）

Department of Physics, Tsinghua University

Office: C406, Phone: 8-4522

Email: shuoyang@tsinghua.edu.cn

WeChat: condmat-ys

Homework

- Try to write a python code for diagonalizing the spin-1/2 AF Heisenberg model on a 4-site ring. Try to implement both U(1) and translational symmetries.

$$H = \sum_{i=0}^{N-1} \mathbf{S}_i \cdot \mathbf{S}_{i+1} = \sum_{i=0}^{N-1} \left(S_i^x S_{i+1}^x + S_i^y S_{i+1}^y + S_i^z S_{i+1}^z \right)$$

$$= \sum_{i=0}^{N-1} \left[\left(S_i^+ S_{i+1}^- + S_i^- S_{i+1}^+ \right) /2 + S_i^z S_{i+1}^z \right]$$

- Try to write a python code for diagonalizing the transverse-field Ising model on a 4-site ring. Try to implement translational symmetry. Show that the eigenvalues agree with the exact solution.

$$H = -J \sum_{i=1}^{N} \left(g\sigma_i^x + \sigma_i^z \sigma_{i+1}^z \right)$$

Please submit your attempt through http://learn.tsinghua.edu.cn.

Deadline is Oct 16.

More detailed instructions will be provided on Oct 11.

Extra points will be given for those who submit before Oct 11.

Hints of homework 1

Get principle basis

- to implement both U(1) and translational symmetries, the first step is to find out the principle configurations in each Sz subspace

- for example, when $N = 4$, the following 4 basis are related to each other by translation

  3 = 0011, 6 = 0110, 9 = 1001, 12 = 1100

  we choose the smallest configuration 3 as the principle configuration

  in another example, the following 2 basis are related to each other by translation

  5 = 0101, 10 = 1010

  we choose the smallest configuration 5 as the principle configuration

- the principle configurations in each Sz subspace should be

  PriC = [[0], [1], [3, 5], [7], [15]]

  namely, we write a code to generate PriC[Nu][j]

  Nu denotes how many 1 in each Sz subspace

  for example, PriC[1][0] = 1, PriC[2][1] = 5

Get properties of principle basis

- we also need to know, after how many translations, the principle configuration comes back to its original state

- The corresponding numbers in each Sz subspace are

  PriR = [[1], [4], [4, 2], [4], [1]]

  namely, we write a code to generate PriR[Nu][j]

  which has one to one correspondence with PriC

  PriC = [[0], [1], [3, 5], [7], [15]]

- for example, PriC[2][1] = 5 and PriR[2][1] = 2

  this means for the configuration 5 = 0101, after 2 translations it comes back to 5 = 0101 again

Get principle basis in each subspace

- to build the Hamiltonian in each Sz and k subspace, we need to check which principle configurations are allowed in each k subspace

- because only when kR_a is a multiple of 2π, the configuration is allowed where $k = \dfrac{2\pi m}{N}, m = 0, 1, \cdots, N-1$
  so that only when mod(m * PriR , N) = 0, the configuration is allowed

- the allowed basis in each [Sz , k] subspace should be
  Basis = [[[0] [] [] []]
  [[1] [1] [1] [1]]
  [[3, 5] [3] [3, 5] [3]]
  [[7] [7] [7] [7]]
  [[15] [] [] []]]

- for example, when Nu = 2 and m = 1, only principle configuration 3 is allowed while 5 is not allowed

Get properties of every configuration

- for a given configuration, we need to know its corresponding principle configuration, and how many translation steps can convert it to the principle configuration

- we may write a code to generate a list Check[i,:]

```
[[ 0  0  0]
 [ 0  1  0]
 [ 1  1  1]
 [ 0  3  0]
 [ 1  1  2]
 [ 0  5  0]
 [ 1  3  1]
 [ 0  7  0]
 [ 1  1  3]
 [ 1  3  3]
 [ 1  5  1]
 [ 1  7  3]
 [ 1  3  2]
 [ 1  7  2]
 [ 1  7  1]
 [ 0 15  0]]
```

- the 0th column denotes whether the configuration is a principle configuration, 0 for yes, 1 for no

  the 1st column gives the principle configuration

  the 2nd column gives the translation steps

- for example, 8 = 1000 is obtained from the principle configuration 1 = 0001 through 3 translation steps

  so that Check[8 , 0] = 1, Check[8 , 1] = 1, Check[8 , 2] = 3

- we may write a code with input N and output PriC, PriR, Check, Basis

Build the Hamiltonian

- if the (Sz , k) subspace is not empty, we can build Hamiltonian on its basis

- the matrix elements of the off-diagonal terms are

$$\langle b_j(k)|\, H_j \,|a(k)\rangle = \sqrt{\frac{N_{b_j}}{N_a}}\, h_j(a)\, e^{-ikl_j}$$

- since $N_a = \frac{N^2}{R_a}$, we have $\sqrt{\frac{N_{b_j}}{N_a}} = \sqrt{\frac{\mathrm{PriR}_a}{\mathrm{PriR}_{b_j}}}$

- given a column number of the matrix, we first find out its corresponding configuration a , then apply one term of the Hamiltonian H_j on a, and get the new configuration b'_j

- using Check, we can know the corresponding new principle configuration b_j and the translation steps $-l_j$

- from the new principle configuration b_j and 'Basis' array, we can know its index in the subspace, which gives the row number of the matrix

Build the Hamiltonian

- for example, when $N = 4$, Nu = 2 and $k = 0$, the matrix elements are

```
Nu= 2 k= 0
0 3 0011 1 2 0101 5 5 0 2 4 1 (0.7071067811865476+0j)
0 3 0011 3 0 1010 10 5 1 2 4 1 (0.7071067811865476+0j)
1 5 0101 0 1 0110 6 3 1 4 2 0 (0.3535533905932738+0j)
1 5 0101 1 2 0011 3 3 0 4 2 0 (0.3535533905932738+0j)
1 5 0101 2 3 1001 9 3 3 4 2 0 (0.3535533905932738+0j)
1 5 0101 3 0 1100 12 3 2 4 2 0 (0.3535533905932738+0j)
```

$|\mathrm{col}\rangle$ $\quad i \quad i+1 \quad$ $\langle b_j|$ $\quad$ $\langle \mathrm{row}|$ $\quad$ value

new principle configuration

$|a\rangle$ principle configuration decimal

$\langle b'_j|$ new configuration decimal

$|a\rangle$ principle configuration binomial

$\langle b'_j|$ new configuration binomial

$-l_j$

PriR_{b_j}

PriR_a