量子物理计算方法选讲 Fall 2023

by Prof. Shuo Yang, Department of Physics, Tsinghua University

- focus on **strongly correlated** many-body system defined on a lattice (discrete)
- some exposure to algorithms and coding techniques
- pave the way for further theoretical study and research
- homework 80%，final project 20%，involvement 20%

main challenge: no exact ansatz, Hilbert space too large (exponential wall, curse of dimensionality); sign problem (in QMC); violate area law (in tensor network); ...

量子物理计算方法选讲 Fall 2023
- Outline
- Chap.0 Introduction
- Chap.1 Exact Diagonalization
- Chap.2 Density Matrix Renormalization Group
  - Traditional DMRG
  - varMPS
  - Infinite Time-Evolving Block Decimation method (iTEBD)
- Chap.3 Quantum Monte Carlo
  - Tranditional Monte Carlo
  - Quantum Monte Carlo - world line QMC
  - Quantum Monte Carlo - stochastic series expansion (SSE)
- Chap.4 Tensor Network State
  - PEPS
  - 2D Tensor Network

Outline

- Introduction (brief review of QM, microscopic lattice models, quantum phase transition, eigenstate decomposition, SVD, Python programming, ...)
- Exact Diagonalization (Lanczos method, time evolution)
- Density Matrix Renormalization Group (many-body entanglement, MPS and symmetry, iTEBD, variational power of MPS)
- Tensor Networks (PEPS, MERA, TRG)
- Quantum Monte Carlo (important sampling, classical MC, fermion sign problem, Determinant QMC, Integral QMC, Variational QMC, *etc*)
- Other methods (NRG, DMFT, ML)
- Summary and Discussions

Chap.0 Introduction

state space

$$|0\rangle = |\uparrow\rangle, \quad |1\rangle = |\downarrow\rangle.$$

$|\rightarrow\rangle$, $|\leftarrow\rangle$ are eigenstates of Pauli-X operator:

$$|\rightarrow\rangle = \frac{1}{\sqrt{2}}(|\uparrow\rangle + |\downarrow\rangle), \quad |\leftarrow\rangle = \frac{1}{\sqrt{2}}(|\uparrow\rangle - |\downarrow\rangle),$$

density matrix, pure state, mixed state

measurement process: measurement operator $\{M_m\}$

probability: $\Pr(m) = \mathrm{Tr}\left(M_m \rho M_m^\dagger\right)$

expectation value: $\langle\psi|O|\psi\rangle$ or $\mathrm{Tr}(O\rho)$

inverse temperature (or imaginary time) $\beta = \frac{1}{k_B T}$

classical statistic:

$$(i, E_i) \quad p_i = \frac{1}{Z}\exp(-\beta E_i) \quad Z = \sum \exp\left(-\beta E_i\right)$$

quantum statistic:

- density matrix at the inverse temperature β: $\rho_\beta = \frac{1}{Z}\exp(-\beta H)$
- partition function $Z = \mathrm{Tr}\exp(-\beta H)$
- thermal average of an observable

$$\langle O\rangle = \frac{1}{Z}\mathrm{Tr}(-\beta H O) = \frac{\mathrm{Tr}(-\beta H O)}{\mathrm{Tr}(-\beta H)}$$

Schrodinger equation:

$$i\partial_t \left|\psi(t)\right\rangle = H\left|\psi(t)\right\rangle$$

stationary time-independent SE:

$$H\left|\psi\right\rangle = E\left|\psi\right\rangle$$

SE for density matrix (for closed system)

$$i\partial_t \rho(t) = [H, \rho(t)]$$

second quantization

- occupation number representation $\left|f_{d-1}, \cdots, f_0\right\rangle$
- bosonic creation & annihilation operators

$$a_p^\dagger \left|f_{d-1}, f_{d-2}, \cdots, f_0\right\rangle = \sqrt{f_p + 1}\left|f_{d-1}, f_{d-2}, \cdots, f_p + 1, \cdots, f_0\right\rangle,$$

$$a_p \left|f_{d-1}, f_{d-2}, \cdots, f_0\right\rangle = \sqrt{f_p}\left|f_{d-1}, f_{d-2}, \cdots, f_p - 1, \cdots, f_0\right\rangle.$$

- fermionic creation & annihilation opertors

$$a_p \left|f_{d-1}, f_{d-2}, \cdots, f_0\right\rangle = \delta_{f_p,1}(-1)^{\sum_{i=0}^{p-1} f_i}\left|f_{d-1}, \cdots, f_p \oplus 1, \cdots, f_0\right\rangle,$$
$$a_p^\dagger \left|f_{d-1}, f_{d-2}, \cdots, f_0\right\rangle = \delta_{f_p,0}(-1)^{\sum_{i=0}^{p-1} f_i}\left|f_{d-1}, \cdots, f_p \oplus 1, \cdots, f_0\right\rangle.$$

- commutation relations for bosons:

$$[a_p^\dagger, a_q^\dagger] = 0, \quad [a_p, a_q] = 0, \quad [a_p, a_q^\dagger] = \delta_{pq}$$

- commutation relations for fermions:

$$\{a_i, a_j\} = \{a_i^\dagger, a_j^\dagger\} = 0, \quad \{a_i, a_j^\dagger\} = \delta_{ij}$$

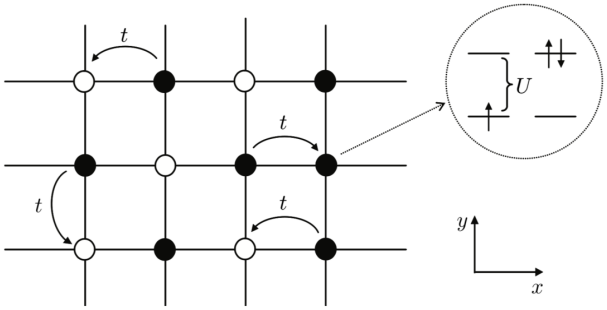
tensor product state (np.kron in numpy)

Microscopic lattice model: (search the Hamiltonian)

- Hubbard model

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

  - 1st term: describe the hopping of electrons between two neighboring sites, called **hopping term**
  - 2nd term: describe the onsite Coulomb repulsion between two electrons occupying the same site



thinking: band structure?

- Heisenberg model

  can be viewed as Hubbard model when $U \gg t$.

$$H = J \sum_{\langle i,j \rangle} \mathbf{S}_i \cdot \mathbf{S}_j = J \sum_{\langle i,j \rangle} (S_i^x S_j^x + S_i^y S_j^y + S_i^z S_j^z)$$

- transverse field Ising model

$$H = -J \sum_{i=1}^{N} \left(g\sigma_i^x + \sigma_i^z \sigma_{i+1}^z \right), \quad J, g \geq 0.$$

  - quantum phase transition

    phase transitions at zero temp., which occur when some parameter in the Hamiltonian is varied.

  - ground state in 2 limits:

$$|G\rangle = \begin{cases} |\uparrow\uparrow \cdots \uparrow\rangle \text{ or } |\downarrow\downarrow \cdots \downarrow\rangle, & g = 0 \\ |\rightarrow \cdots \rightarrow\rangle, & g \to \infty \end{cases}$$

Chap.1 Exact Diagonalization

- why exact diagonlization?
  - **complete and accurate** knowledge of a quantum system can be obtained.
  - **insights** gained from ED are very useful.
  - indispensable for **testing the correctness and benchmarking** other algorithms
  - ED provides a concrete path for learning many important aspects of QM (in particular, the **symmetry** properties of many-body states)
- Since the exponentially-scaling Hilbert space, exact diagonlization is limited to rather small spins (examples):
  - Fermionic models (Hubbard)
  - full configuration interaction (FCI) in quantum chemistry and nuclear structure
  - quantum field theory (QFT)
- Present day limits of ED:
  - spin $S - 1/2$ models:

    $N \sim 30 - 40$
  - Hubbard models:

    $N \sim 20$
- Coding
  - Hilbert space (basic representation (real space or moment space), lookup techniques, symmetries (reduce to small blocks))
  - Hamiltonian matrix (sparsity, matrix recalculation on the fly)
  - linear algebra (eigensolver / time propagation, LAPACK full diagonalization, Lanczos type diagonalization (needs only operations))
  - observables
- **Bit operations in python**

  (see the lecture notes)

- $$H=\sum_{i=0}^{N-1}\mathbf{S}_i\cdot\mathbf{S}_{i+1}=\sum_{i=0}^{N-1}[(S_i^+S_{i+1}^-+S_i^-S_{i+1}^+)/2+S_i^zS_{i+1}^z]$$

  diagonal&off-diagonal terms

- Symmetries
  - $\mathrm{U}(1)$ related symmetries: conservation of **particle numbers** or total S_z
  - translation symmetry (moment space)
  - parity symmetry, reflection symmetry
  - spin-inversion symmetry
  - $\mathrm{SU}(2)$ symmetry
  - various spatial symmetry
- Example: implement of $\mathrm{U}(1)$ symmetry:

  consider the subspaces spanned by **these basis states** with a **conversed up spin** or particle numbers

  for example, a lattice model with 4 spin sites, state with up spin$=2$ can only live in a $\dim=C_4^2=6$ subspace: $\mathrm{Span}(|0011\rangle,|0101\rangle,|0110\rangle,|1001\rangle,|1010\rangle,|1100\rangle)$
- Real-time evolution (RTE)

  $$|\psi(t)\rangle=e^{-iHt}|\psi(0)\rangle=\sum_n e^{-iE_nt}|\psi_n\rangle\langle\psi_n||\psi(0)\rangle=\sum_n C_ne^{-iE_nt}|\psi_n\rangle.$$

  where

  $$C_n=\langle\psi_n|\psi(0)\rangle$$

  for efficient computation of RTE, we can use the Lanczos vectors.
- Non-Hermitian Hamitonian

  $$H|F_n\rangle=\omega_n|F_n\rangle,\quad \langle G_n|H=\langle G_n|\omega_n.$$

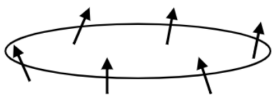
  (bi-orthogonal basis):

  $$I=\sum_n\frac{|F_n\rangle\langle G_n|}{\langle G_n|F_n\rangle}.$$

  question: non-hermitian time-evolution?
- Implementing symmetry
  - last lecture: symmetries about spin and particle-number preservation
  - this lecture: **translation (momentum space),** spatial, SU_2...
- Dispersion relation
  - Pauli Dirac: relativistic dispersion relation:

    $$E^2=(mc^2)^2+(pc)^2$$
  - ultra-relativistic limit: $E=pc$

    non-relativistic limit: $E=p^2/2m$
  - solid physics: collection of all possible energies and momenta is known as the **band structure** of this material (e.g., insulator, semiconductor, conductor)
- Translational symmetry



  - projector: $P_k=\frac{1}{N}\sum_{j=0}^{N-1}e^{i\frac{2\pi}{N}kj}T^j,\quad \text{where} j=0,\cdots,N-1$
  - $\{P_k|r\rangle\}$ is eigenstates of T, with corresponding eigenvalues e^{-iK}, where $K=\frac{2\pi k}{N}$ is the discrete lattice momentum
  - $P_k^\dagger=P_k,\ P_k^2=P_k$

-

	$\lvert r\rangle$	$T^0\lvert r\rangle$	$T^1\lvert r\rangle$	$T^2\lvert r\rangle$	$T^3\lvert r\rangle$	$P_{k=0}\lvert r\rangle$	$P_{k=1}\lvert r\rangle$	$P_{k=2}\lvert r\rangle$	$P_{k=3}\lvert r\rangle$
$\#(\uparrow)=0$	$\lvert 0000\rangle$	$\lvert 0000\rangle$	$\lvert 0000\rangle$	$\lvert 0000\rangle$	$\lvert 0000\rangle$	✓	×	×	×
$\#(\uparrow)=1$	$\lvert 0001\rangle$	$\lvert 0001\rangle$	$\lvert 0010\rangle$	$\lvert 0100\rangle$	$\lvert 1000\rangle$	✓	✓	✓	✓
$\#(\uparrow)=2$	$\lvert 0011\rangle$	$\lvert 0011\rangle$	$\lvert 0110\rangle$	$\lvert 1100\rangle$	$\lvert 1001\rangle$	✓	✓	✓	✓
	$\lvert 0101\rangle$	$\lvert 0101\rangle$	$\lvert 1010\rangle$	$\lvert 0101\rangle$	$\lvert 1010\rangle$	✓	×	✓	×
$\#(\uparrow)=3$	$\lvert 0111\rangle$	$\lvert 0111\rangle$	$\lvert 1110\rangle$	$\lvert 1101\rangle$	$\lvert 1011\rangle$	✓	✓	✓	✓
$\#(\uparrow)=4$	$\lvert 1111\rangle$	$\lvert 1111\rangle$	$\lvert 1111\rangle$	$\lvert 1111\rangle$	$\lvert 1111\rangle$	✓	×	×	×

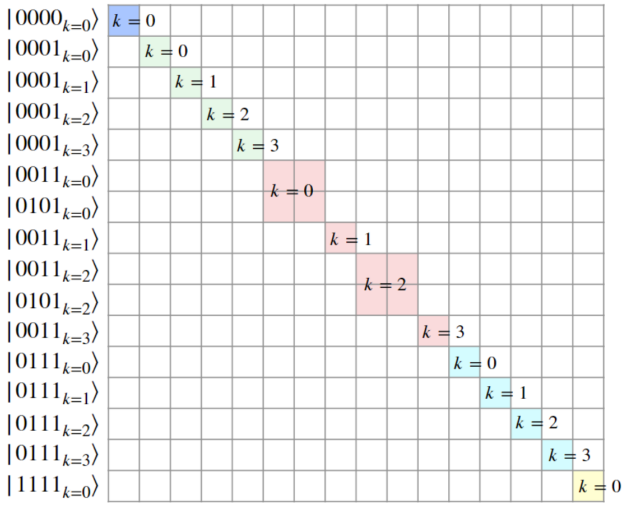
  representative basis; what is the corresponding representative configuration of any basis vector; the translation step

  (construct an array "Check")

- Normalization and Hamiltonian
  - $\lvert r_k\rangle = \frac{P_k\lvert r\rangle}{\sqrt{\langle r\rvert P_k^\dagger P_k\lvert r\rangle}} = \frac{P_k\lvert r\rangle}{\sqrt{\langle r\rvert P_k\lvert r\rangle}}$
  - discard those $\lvert r\rangle$ with $P_k\lvert r\rangle = 0$
  - since $[T, H] = 0$, we have $[P_k, H] = 0$
  - Since H is already blockwise diagonal under $\{\lvert r_k\rangle\}$, $\langle r'_{k'}\rvert H\lvert r_k\rangle$ must be zero for $k \neq k'$
  - $$\langle r'_k\rvert H\lvert r_k\rangle = \frac{\langle r'\rvert P_k H\lvert r\rangle}{\sqrt{\langle r'\rvert P_k\lvert r'\rangle\langle r\rvert P_k\lvert r\rangle}}.$$

    where, $\langle r\rvert P_k\lvert r\rangle$ can be obtained from calculation of $P_k\lvert r\rangle$ (storing the coefficients)

    we may also need to look up the table to calculate $(P_k\lvert r'\rangle, H\lvert r\rangle)$ (can be non-zero only when representative state matches $H\lvert r_k\rangle$)



  - When we try to implement translational symmetry on fermionic systems, but sometimes coding is a headache (T may introduce additional sign)
    - apply FT to the whole Hamiltonian (into momentum space)
    - get matrix representation of T, diagonalize T to get the momentum basis

- Example: Exact solution of transverse field Ising model
  - Jordan-Wigner mapping: very powerful mapping between spin-1/2 spins and spineless fermions
  - Fourier transformation (momentum space)
  - Bogliubov transformation (based on momentum space, non-interacting single particle)
- Iterative diagonalization: variational principle

$$E[|\psi\rangle] = \frac{\langle\psi|H|\psi\rangle}{\langle\psi|\psi\rangle}$$

is minimized for $\psi = \psi_0$ with $E[\psi_0] = E_0$.

- Power method

  $$|v_n\rangle = H^n |v_0\rangle$$

  expanding in the eigenbasis:

  $$|v_n\rangle = H^n \left[\sum_i |i\rangle\langle i|\right] |v_0\rangle = \sum_i \langle i|v_0\rangle \lambda_i^n |i\rangle$$

  the state with the eigenvalue with the largest absolute value will have the highest weight after many iterations n

  provided that $|v_0\rangle$ has a finite overlap with this state

- very simple to implement memory efficient, only two vectors must be stored in memory
- Krylov method

  $$\{|v_0\rangle, H|v_0\rangle, \cdots, H^n|v_0\rangle\}$$

  nth Krylov space

  an optimal linear combination of vectors approximating the extremal eigenstate exists, and the way to find it is to diagonalize H in the subspace generated by $n+1$ vectors

  (基本可认为能找到多个准确的低激发态)

  - start from $|\phi_0\rangle$ (only need it has non-vanishing overlap of the true ground state of H)
  - our aim: write H in as a tridiagonal matrix on a set of basis $\{|\phi_0\rangle, \cdots, |\phi_n\rangle\}$
  - number of Lanczos vectors needed: in the order of **few tens to hundreds**
  - energies often converge **faster** than other observables
- drawback of Krylov method: loss of generality after long iterations (sometimes leading to fake eigenvalues)
  - straightforward solution: reorthogonalize Lanczos vectors relative to each other using a modified Gram-Schmidt procedure (requires all vectors to be stored in memory, so that the advantage of memory efficiency lost)
  - or: start from the same initial state and re-calculate all vectors (after 10 to 100 Lanczos iterations, the resulting tridiagonal matrix is **diagonalized** and the **extremal eigenstate is used as starting vector** for a new Lanczos procedure)

    (implicitly Lanczos method)
- Green function
  - inhomogeneous linear pde:

    $$Df(x) = g(x)$$
  - particular solution:

    $$f(x) = \int G(x, x')g(x')\mathrm{d}x'$$

    the Green's function is defined as the solution of a differiential equation:

    $$DG(x, x') = \delta(x, x')$$
  - SE in real space:

    $$\left[i\partial_t + \frac{1}{2}\triangle\right] G(r, t; r', t') = \delta(r - r')\delta(t - t')$$

    $$\Psi(r, t) = \int G(r, t; r', t')V(r', t')\Psi(r', t')\,\mathrm{d}^3\,r'$$

    Green's function is known as the propagator

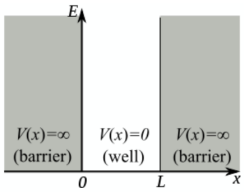
Chap.2 Density Matrix Renormalization Group

Traditional DMRG

the key is to truncate the Hilbert space

challenge: select a basis that will work **in general**

- 1975 NRG (数值重整化群)
- 1992 DMRG (密度矩阵重整化群)
- 1D particle-in-a-box
  - $H\psi(x) = -\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}x^2}\psi(x) + V(x)\psi(x) = E\psi(x)$
    where $V(x)$ is:



  - 离散化:
    - 有限差分$D^2\psi(x) \approx \frac{1}{a^2}[\psi(x-a) - 2\psi(x) + \psi(x+a)]$
    - numbering the partition points: $H = -\sum_{l=1}^{L-1}(|l\rangle\langle l+1| + |l+1\rangle\langle l|) + 2\sum_{l=1}^{L}|l\rangle\langle l|$. (**tridiagonal**)
      e.g. $L = 5$:

$$H = \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & 2 & -1 & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}.$$

  - NRG method

$$H = \left(\begin{array}{cc|cc|c} H & T & \ddots & \cdots & 0 \\ T^\dagger & H & T & \ddots & \vdots \\ \hline \ddots & T^\dagger & H & T & \ddots \\ \vdots & \ddots & T^\dagger & H & \ddots \\ \hline 0 & \cdots & \ddots & \ddots & \ddots \end{array}\right) = \left(\begin{array}{cc|c} H' & T' & 0 \\ \hline T'^\dagger & H' & \ddots \\ 0 & \ddots & \ddots \end{array}\right) \to \left(\begin{array}{cc|c} H'' & T'' & 0 \\ \hline T''^\dagger & H'' & \ddots \\ 0 & \ddots & \ddots \end{array}\right)$$

    - $H' = \begin{pmatrix} H & T \\ T^\dagger & H \end{pmatrix}$, $T' = \begin{pmatrix} 0 & 0 \\ T & 0 \end{pmatrix}$ (the size of T needn't to be 1×1, we can consider "blockwise" tridiagonal matrix)
    - diagonalize H', getting eigenvectors V_l, $H'V_n = E_nV_n$
    - build matrix O using **low-energy eigenvectors** (project to low-lying subspace)

$$O = \begin{pmatrix} | & | & & | \\ V_1 & V_2 & \cdots & V_m \\ | & | & & | \end{pmatrix}$$

    - change of basis, truncate

$$H'' = O^\dagger H' O, \quad T'' = O^\dagger T' O$$

$$H'' = \boxed{O^\dagger}\ H'\ \boxed{O}$$

$$T'' = \boxed{O^\dagger}\ T'\ \boxed{O}$$

    - replace H and T by H'' and T'', truncate

- programming: recall `np.argsort` for finding low-lying eigenvectors.
- **NRG fails**: particle in a box eigenstates are 0 at the edges, any state formed by low-lying states has a kink in the middle to remove kink, need to keep almost all states

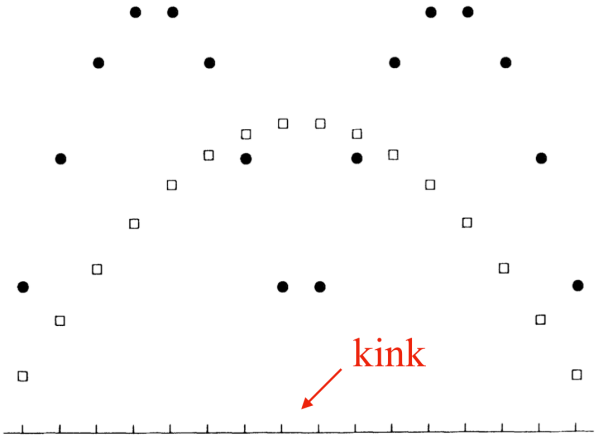


FIG. 1. Lowest eigenstates of two 8-site blocks (solid circles) and a 16-site block (open squares) for the one-dimensional tight-binding model with fixed boundary conditions.

- consider **fixed-free combination**

	free	fixed
free	$H_{11}^{\prime[n+1]} = \begin{pmatrix} H_{12}^{[n]} & T^{[n]} \\ T^{\dagger[n]} & H_{21}^{[n]} \end{pmatrix}, H_{11}^{\prime[1]} = \left(\begin{array}{cc\|cc} 1 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ \hline 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{array}\right)$	$H_{12}^{\prime[n+1]} = \begin{pmatrix} H_{12}^{[n]} & T^{[n]} \\ T^{\dagger[n]} & H_{22}^{[n]} \end{pmatrix}, H_{12}^{\prime[1]} = \left(\begin{array}{cc\|cc} 1 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ \hline 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 2 \end{array}\right)$
fixed	$H_{21}^{\prime[n+1]} = \begin{pmatrix} H_{22}^{[n]} & T^{[n]} \\ T^{\dagger[n]} & H_{21}^{[n]} \end{pmatrix}, H_{21}^{\prime[1]} = \left(\begin{array}{cc\|cc} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ \hline 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{array}\right)$	$H_{22}^{\prime[n+1]} = \begin{pmatrix} H_{22}^{[n]} & T^{[n]} \\ T^{\dagger[n]} & H_{22}^{[n]} \end{pmatrix}, H_{22}^{\prime[1]} = \left(\begin{array}{cc\|cc} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ \hline 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 2 \end{array}\right)$

free,free ; free,fixed ; fixed,free ; fixed,fixed

$$T^{\prime[n+1]} = \begin{pmatrix} 0 & 0 \\ T^{[n]} & 0 \end{pmatrix}, T^{\prime[1]} = \left(\begin{array}{cc\|cc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{array}\right)$$

use the lowest $m/4$ states from each of H_{11},H_{12}, H_{21}, H_{22}

transforming matrix:

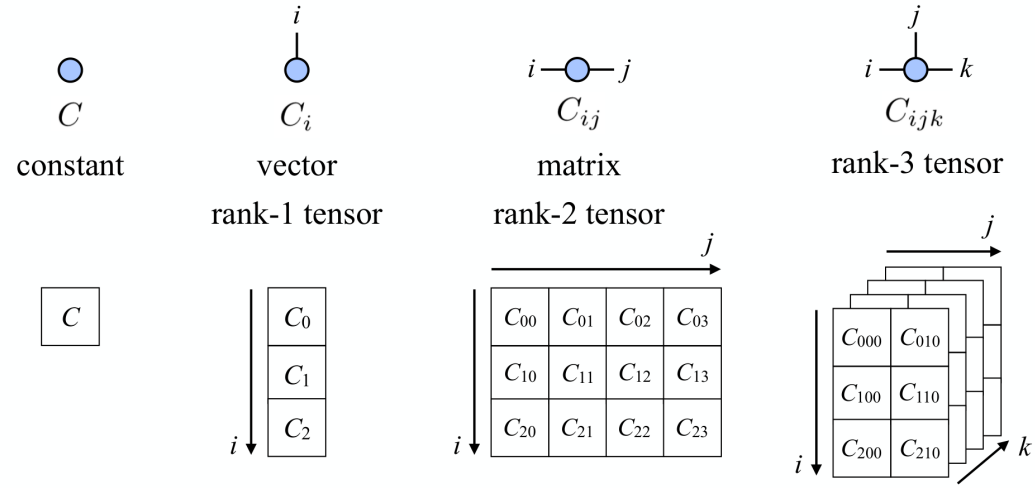
$$\widetilde{O} = \begin{pmatrix} | & | & & | & | & | & & | & | & | & & | & | & | & & | \\ V_1^{11} & V_2^{11} & \cdots & V_{m/4}^{11} & V_1^{12} & V_2^{12} & \cdots & V_{m/4}^{12} & V_1^{21} & V_2^{21} & \cdots & V_{m/4}^{21} & V_1^{22} & V_2^{22} & \cdots & V_{m/4}^{22} \\ | & | & & | & | & | & & | & | & | & & | & | & | & & | \end{pmatrix}$$

$$O = \text{Gram} - \text{Schimidt}(\widetilde{O}), \quad \text{range}(O) = \text{range}(\widetilde{O})$$

$$H_{11}^{[n]} = O^\dagger H_{11}^{\prime[n]} O, \quad H_{12}^{[n]} = O^\dagger H_{12}^{\prime[n]} O, \quad H_{21}^{[n]} = O^\dagger H_{21}^{\prime[n]} O, \quad H_{22}^{[n]} = O^\dagger H_{22}^{\prime[n]} O, \quad T^{[n]} = O^\dagger T^{\prime[n]} O$$

- Tensor network

  Roger Penrose's graphical notation

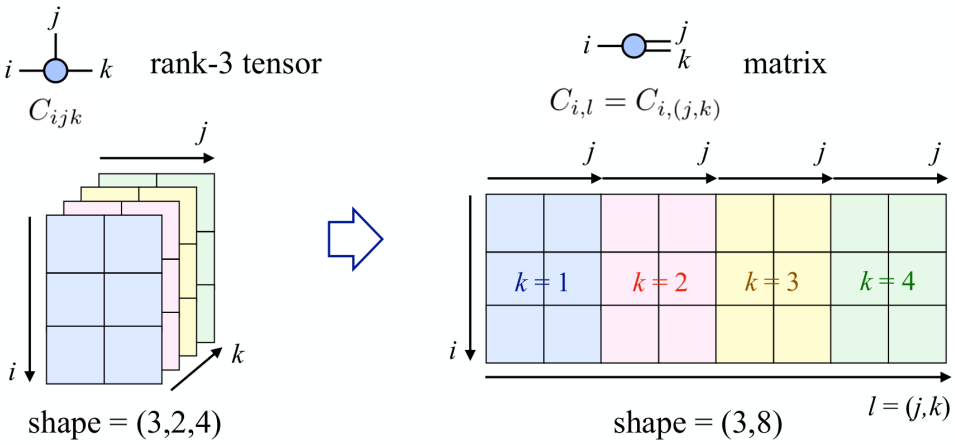


- an N-spin quantum state

$$|\psi\rangle = \sum_{i_1 i_2 \cdots i_N} C_{i_1 i_2 \cdots i_N} |i_1 i_2 \cdots i_N\rangle$$

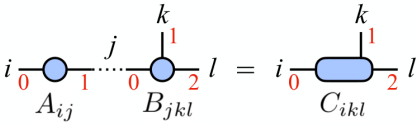
is a rank-N tensor

- Reshape a tensor



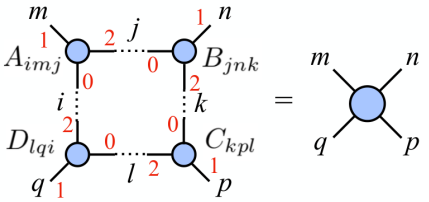

```
import numpy as np
C = np.random.rand(3,2,4)
C = np.reshape(C,[3,8])
np.shape(C)
>>> (3,8)
```

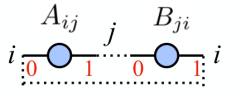
- Tensor contraction
  - connecting 2 index lines implies a **contraction** (summation over the same indices)



$$\sum_j A_{ij} B_{ijkl} = C_{ikl}$$

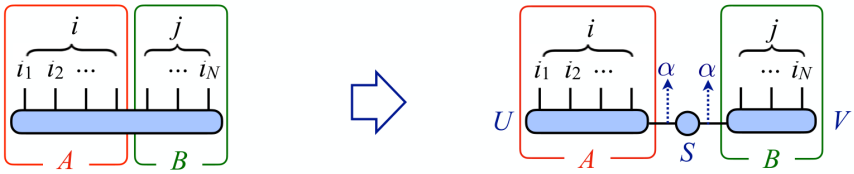
    - `np.einsum`
    - `np.tensordot(A,B,(1,0))` (contract the 1st index of A and the 0th index of B)
  - some examples:





$$\sum_{ij} A_{ij}B_{ji} = \mathrm{Tr}\,(AB)$$

- Schmidt decomposition
  - physical meaning: bipartate a quantum system in blocks A and B, with orthogonal bases



  - bipartate the indices into i and j

$$|\psi\rangle = \sum_{i_1,\cdots,i_N} c_{i_1,\cdots,i_N} |i_1,\cdots,i_N\rangle = \sum_{i,j} c_{ij} |i\rangle_A |j\rangle_B$$

    apply SVD on c_{ij} we have:

$$|\psi\rangle = \sum_{i,j} \left[\sum_\alpha \sigma_\alpha u_i^\alpha v_j^\alpha\right] |i\rangle_A |j\rangle_B = \sum_\alpha \sigma_\alpha |u^\alpha\rangle_A |v^\alpha\rangle_B$$

    - This procedure is called Schmidt's decomposition
    - (u_i^α) is left-isometric ($U^\dagger U = \mathrm{Id}$), (v_j^α) is right-isometric ($VV^\dagger = \mathrm{Id}$)

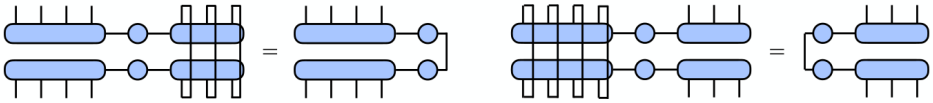
      $|u^\alpha\rangle_A$ and $|v^\alpha\rangle_B$ form orthogonal bases

      graphical notation of the orthogonal condition (left-canonical and right-canonical)

  - reduce density matrix of a bipartate quantum system:
    - Schmidt decomposition
    - density matrix of the whole system: $\rho = |\psi\rangle\langle\psi|$
    - reduced density matrix: **partial trace** ($\rho_A = M_A \rho M_A^\dagger$, where $M_A = \mathrm{Proj}_A \otimes \mathrm{Id}_B$) of ρ

      $\rho_A = \mathrm{Tr}_B\,\rho = \sum_\alpha s_\alpha^2 |u^\alpha\rangle\langle u^\alpha|_A$

      $\rho_B = \mathrm{Tr}_A\,\rho = \sum_\alpha s_\alpha^2 |v^\alpha\rangle\langle v^\alpha|_B$



    - Entanglement entropy:

$$S = -\,\mathrm{Tr}(\rho_A \log \rho_A) = -\,\mathrm{Tr}(\rho_B \log \rho_B) = -\sum_\alpha s_\alpha^2 \log s_\alpha^2.$$

【Example】tensor product state

$$|\psi\rangle = \frac{1}{2}(|\uparrow\rangle_A + |\downarrow\rangle_A)(|\uparrow\rangle_B + |\downarrow\rangle_B).$$

$$\rho_A = \frac{1}{2}\begin{pmatrix}1 & 1\\ 1 & 1\end{pmatrix} = \left[\frac{1}{\sqrt{2}}\begin{pmatrix}1 & -1\\ 1 & 1\end{pmatrix}\right]\begin{pmatrix}1 & 0\\ 0 & 0\end{pmatrix}\left[\frac{1}{\sqrt{2}}\begin{pmatrix}1 & -1\\ 1 & 1\end{pmatrix}\right]^\dagger, \quad S = -1\cdot\log 1 - 0\cdot\log 0 = 0.$$

(product state has no entanglement)

【Example】maximally entangled state

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|\uparrow\rangle|\downarrow\rangle + |\downarrow\rangle|\uparrow\rangle)$$

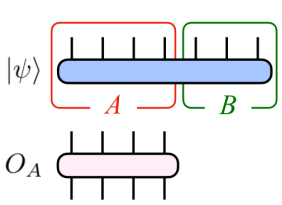
$$\rho_A = \frac{1}{2}\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}, \quad S = \log 2.$$

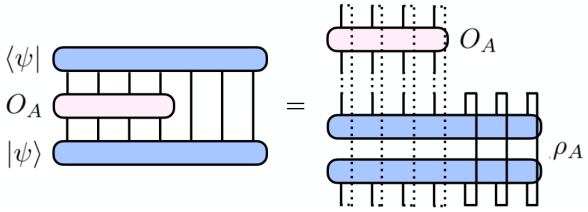
(for d dimensional bipartate system, its maximal entanglement entropy will be $\log d$)

- why can we truncate on rdm?
  - given $|\psi\rangle$ and an operator O_A acting on A part, the expectation value is

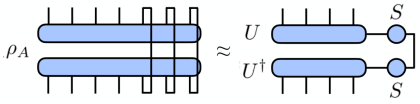
    $$\langle O_A \rangle = \langle\psi| O_A |\psi\rangle = \mathrm{Tr}(O_A \rho_A)$$

    graphically representation:

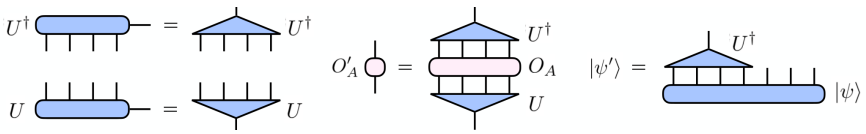
    

  - the reduced density matrix ρ_A tells us "**which states are the most important**" (truncate on S^2)

    $$\rho_A \approx U S^2 U^\dagger$$

  - use $U^\dagger$ and U to truncate the operator O_A:

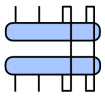
    $$O'_A = U^\dagger O_A U, \quad |\psi'\rangle = (U^\dagger \otimes I_B)\,|\psi\rangle.$$

    (equivalent to NRG !)

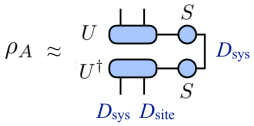
-
- Infinite DMRG algorithm
  - form Hamiltonian matrix for the superblock

    (system, site, site, environment)
  - diagonalize the superblock Hamiltonian to find the ground state $|\psi\rangle$

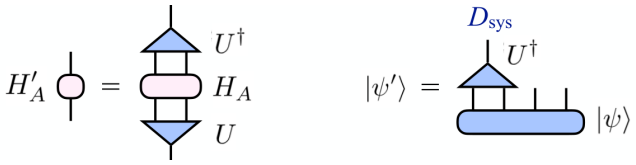
    
  - form the rdm of (system+site) ρ_A

    
  - diagonalize ρ_A to find U and S and make a truncation to U and S

    $D_{\text{sys}} \times D_{\text{site}} \to D_{\text{site}}$.

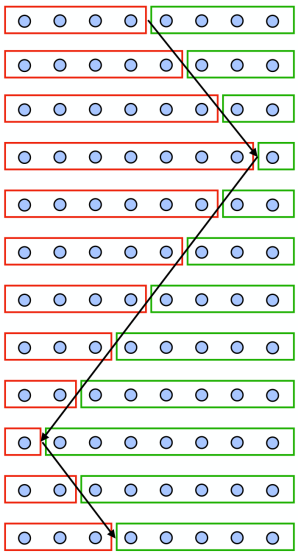
    

  - form the new system block using U

  - Sample code of traditional DMRG https://github.com/simple-dmrg/simple-dmrg
- Finite DMRG algorithm

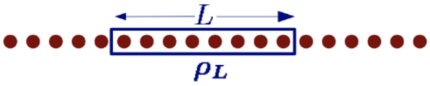
  sweep back and forth until convergence is reached

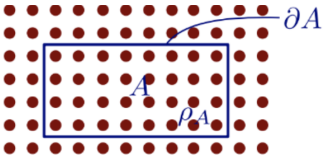


MPS & MPO

- Physical corner of Hilbert space
  - $|\psi\rangle = \sum_{i_1,\cdots,i_N} c_{i_1,\cdots,i_N} |i_1,\cdots,i_N\rangle$, exponentially large Hilbert space
  - local Hamiltonian has only $O(N)$ params
  - ground state must live in a **small physical corner** of Hilb. space (can be described using less params using entanglement structure)
- Recall many-body entanglement
  - Schmidt decomposition $|\psi\rangle = \sum_\alpha s_\alpha |u_\alpha\rangle_A \otimes |v_\alpha\rangle_B$
  - reduced density matrix $\rho_A = \mathrm{Tr}_B\, \rho$
  - EE $S = -\mathrm{Tr}(\rho_A \log \rho_A) = -\mathrm{Tr}(\rho_B \log \rho_B) = -\sum_\alpha s_\alpha^2 \log s_\alpha^2$.
  - maximum possible entanglement entropy: $\log D$ ($\mathrm{schmidt\ rank} = D$)
- **Area Law**
  - entropy of reduced states scales like **the length of its boundary** ∂A

    if L is the spins in the reduced states, we calculate its bipartite entanglement entropy:

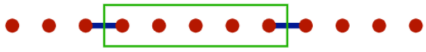
    

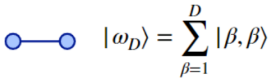
    gapped 1D system ground state: $S(\rho_L) \sim \mathrm{const.}$

    gapless 1D system ground state: $S(\rho_L) \sim \log L$

    physical explanation: the entanglement between two regions is located around the boundary

    
- constructing MPS ansatz

  our goal: construct an ansatz for quantum many-body systems which **satisfies area law**
  - suppose each site is composed of two virtual sites
  - virtual sites are placed in **maximally entangled states** $|\omega_D\rangle$, area law is automatically satisfied:

  - **map virtual sites to physical sites by the operator** P**:**

$$P_i = \sum_{i,\alpha,\beta} A^{[i]}_{i,\alpha,\beta} |i\rangle\langle\alpha,\beta|.$$

(the coefficients $A^{[i]}_{i,\alpha,\beta}$ are to be determined, we called the DOF α, β virtual DOF, and i physical DOF)

Therefore, we can construct the total state as:

$$|\psi\rangle = \bigotimes_{i=1}^{N} P_i |\omega_D\rangle^{\otimes N}$$

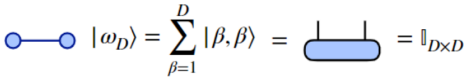
other perspective:

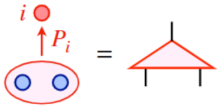
each site is described using a 3-order tensor A ($2ND^2$ parameters in total, compared to 2^N), the total state is a **contraction** of these 3-order tensor.

Under this perspective,

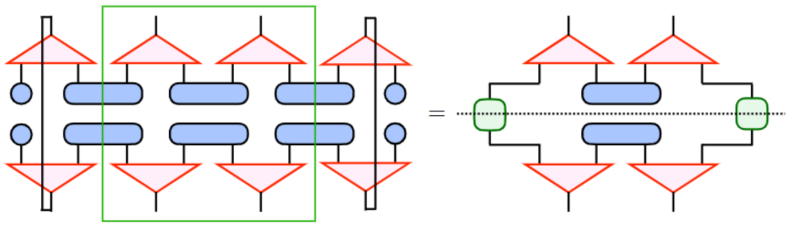
$$|\omega_D\rangle = \sum_{\beta=1}^{D} |\beta\beta\rangle = \mathrm{Id}_{D\times D}.$$

tensor network representation:





reduced density matrix: tracing out the green framed part:



we can see easily the maximum possible EE is $2\log D$, area law is satisfied.

- expression of MPS ansatz:

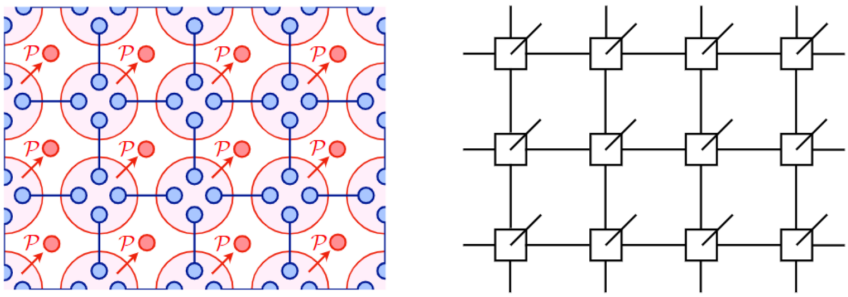
  two-site example:

$$(P_1 \otimes P_2)\,|\omega_D\rangle = \left[\sum_{i_1\alpha\beta} A^1_{i_1\alpha\beta}\,|i_1\rangle\langle\alpha\beta|\right]\left[\sum_{i_2\beta\gamma} A^2_{i_2\beta\gamma}\,|i_2\rangle\langle\beta\gamma|\right]\left[\sum_{\beta}|\beta\beta\rangle\right] = \sum_{i_1 i_2\alpha\beta\gamma}\left(A^1_{i_1\alpha\beta}A^2_{i_2\beta\gamma}\right)|i_1 i_2\rangle\langle\alpha\gamma|.$$

  1D-ring example:

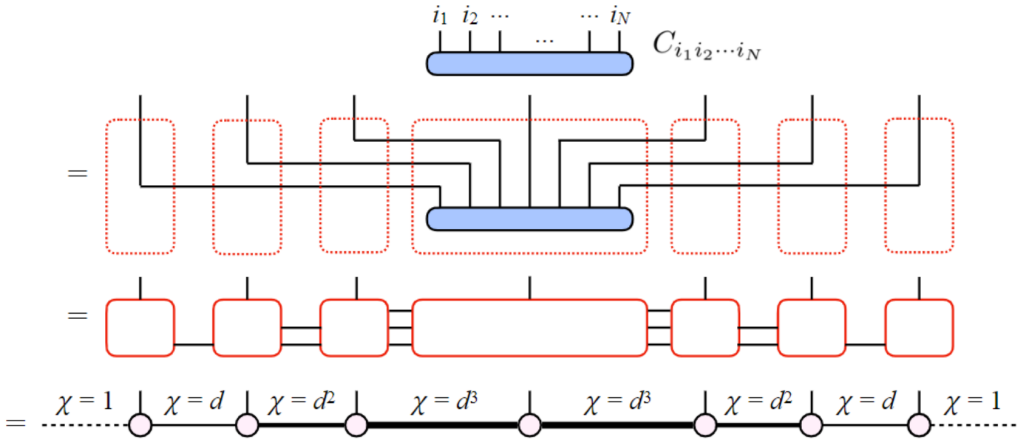
$$|\Psi\rangle = \sum_{i_1,\cdots,i_N} c_{i_1,\cdots,i_N}\,|i_1,\cdots,i_N\rangle = (P_1\otimes\cdots\otimes P_N)|\omega_D\rangle^{\otimes N} = \sum_{i_1,\cdots,i_N}\mathrm{Tr}\left[A^{[1]}_{i_1}A^{[2]}_{i_2}\cdots A^{[N]}_{i_N}\right]|i_1,\cdots i_N\rangle$$

  - Generalization: 2D PEPS (projected entangled-pair states)



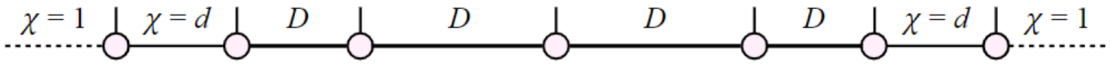
  - FACT: given infinitely large bond dimensions, every state can be written as an MPS:

    Proof:



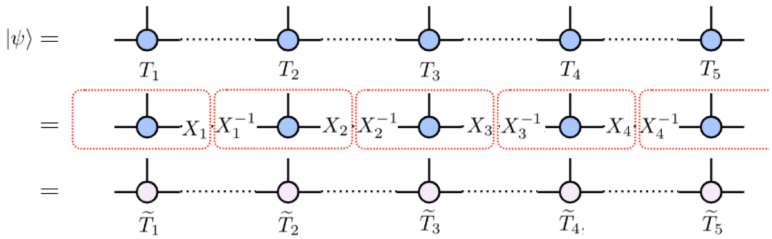
In practice, we can have an appropriate state by choosing a finite bond dimension $D < \max(\chi)$, e.g.,

$$|\psi\rangle \approx$$



and $D \to \infty$ is exact

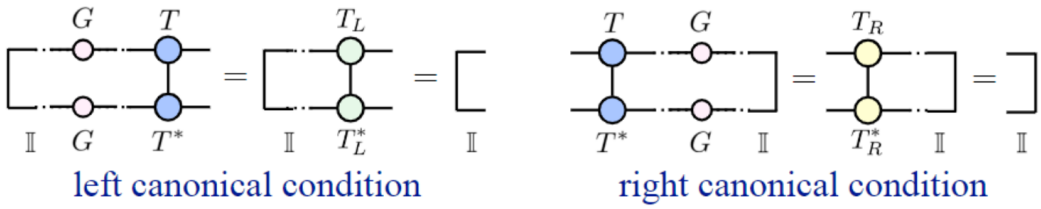
- canonical form of MPS
  - MPS is not unique (up to inserting an additional matrix and its inverse)



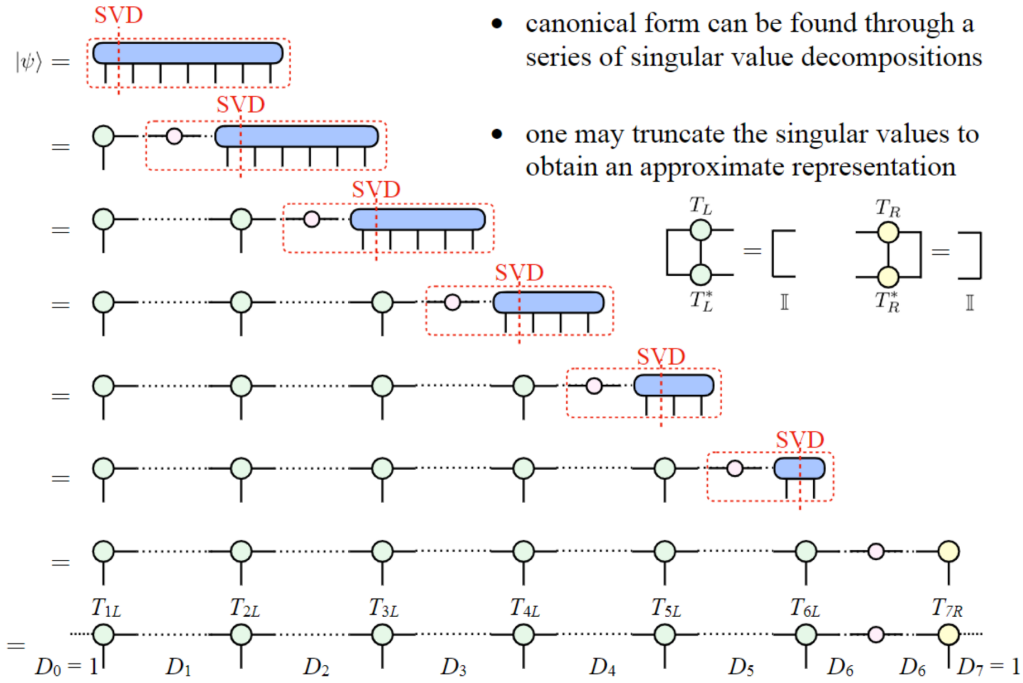
  - canonical form of MPS:

    MPS that satisfies:

    - the gauge degree of freedoms are fixed to correspond with the Schmidt decomposition
    - left and right canonical condition:



    can be found using SVD:

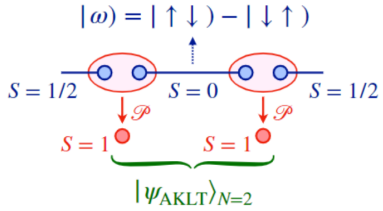


- canonical form can be found through a series of singular value decompositions
- one may truncate the singular values to obtain an approximate representation

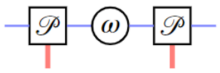
the SVD-based construction of canonical form: can be used to **optimally** truncate the bond-dimension of a quantum state

- Textbook example: **AKLT state** (Affleck, Kennedy, Lieb, and Tasaki, PRL 1987)

  AKLT: has a non-trivial constraint arising from the virtual-spin based AKLT construction



  - AKLT state as MPS:



    $|\omega\rangle = |\uparrow\downarrow\rangle - |\downarrow\uparrow\rangle$

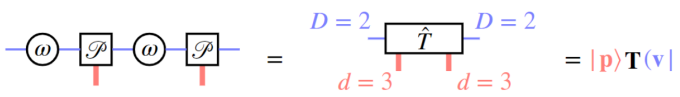
    $\mathscr{P} = |+\rangle\langle\uparrow\downarrow| + |0\rangle\frac{\langle\uparrow\downarrow|+\langle\downarrow\uparrow|}{\sqrt{2}} + |-\rangle\langle\downarrow\downarrow|$

  - parent Hamiltonian H of AKLT state:

$$H = \sum_i \Pi_i, \quad \Pi_i |\psi_{AKLT}\rangle_{N=2} = 0, \quad H\,|\psi_{AKLT}\rangle = 0.$$

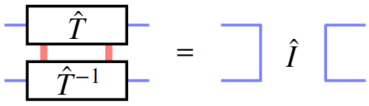
    Can we construct H using MPS?

$$|\psi_{AKLT}\rangle_{N=2} =$$

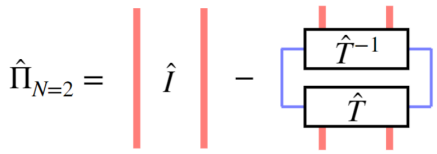


    Recall: dimension of physical index: 3, virtual index: 2

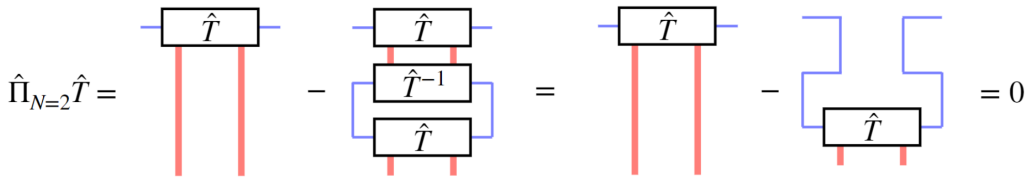
    - T: a map from the $|\mathbf{v}\rangle$ space to the $|\mathbf{p}\rangle$ space
    - left inverse of T:



    - local parent Hamiltonian:



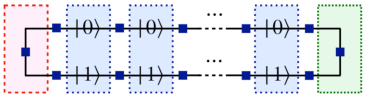
verification:



- Non-Hermitian Parent Hamiltonian (PRL 130, 220401 (2023))

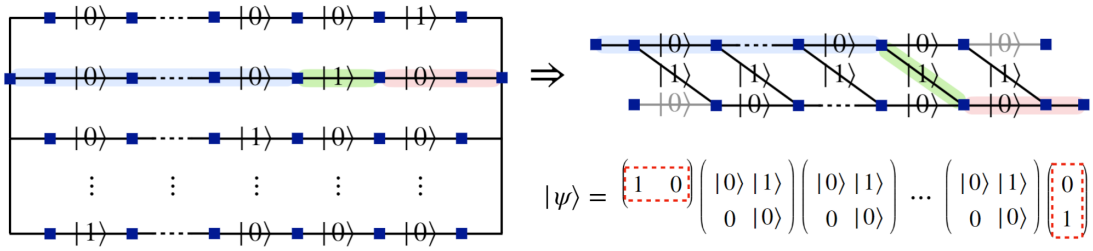
- States with exact matrix product form
  - GHZ state:

$$|\psi\rangle = (1 \quad 1)\begin{pmatrix}|0\rangle & \\ & |1\rangle\end{pmatrix} \otimes \begin{pmatrix}|0\rangle & \\ & |1\rangle\end{pmatrix} \otimes \cdots \otimes \begin{pmatrix}|0\rangle & \\ & |1\rangle\end{pmatrix}\begin{pmatrix}1\\1\end{pmatrix}$$

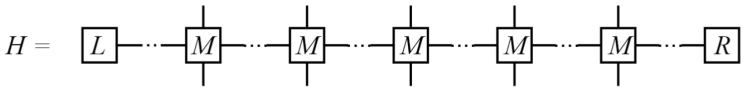


  - W state:

$$|\psi\rangle = |0\cdots01\rangle + |0\cdots010\rangle + \cdots + |1\cdots000\rangle$$



- MPO (Matrix Product Operator)
  - open boundary condition



upper and down: physical bonds

left and right: virtual bonds

  - construct MPO of general Hamiltonian: SVD + contraction
  - construct MPO of simple Hamiltonian: graph method

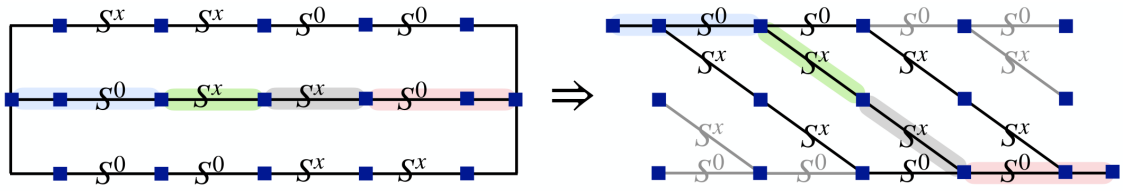
regard M as a matrix:

MPO $\rightarrow$ matrix elements are physical operators

Example:

$$H = \sum_j S_j^x S_{j+1}^x, \quad (N=4)$$

How to generalize these 3 terms? Write as "finite" state automata:



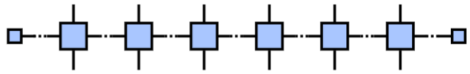
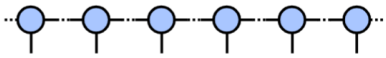
then we can read the component tensors directly from the graph.

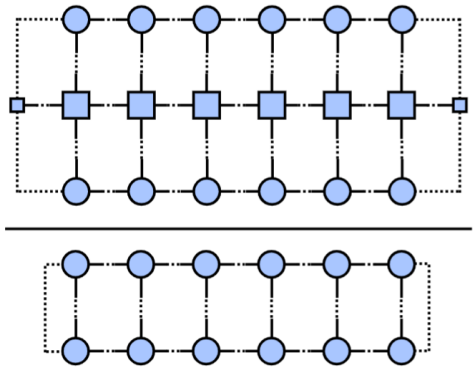
$$H = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} S^0 & S^x & 0 \\ 0 & 0 & S^x \\ 0 & 0 & S^0 \end{pmatrix} \begin{pmatrix} S^0 & S^x & 0 \\ 0 & 0 & S^x \\ 0 & 0 & S^0 \end{pmatrix} \begin{pmatrix} S^0 & S^x & 0 \\ 0 & 0 & S^x \\ 0 & 0 & S^0 \end{pmatrix} \begin{pmatrix} S^0 & S^x & 0 \\ 0 & 0 & S^x \\ 0 & 0 & S^0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

varMPS

- Setting: given a Hamiltonian, we may find its ground state energy by minimizing its energy

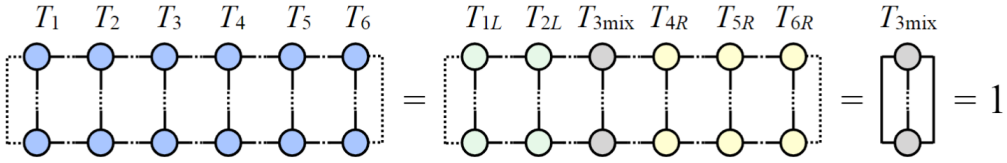
  E.G. 1D system with OBC:

  $H =$  $|\psi\rangle =$ 

  $E = \dfrac{\langle\psi| H |\psi\rangle}{\langle\psi|\psi\rangle} =$ 

  the little square on the left: can be viewed as having a 1-dimensional index

- If we use the canonical form of MPS:

  $\langle\psi|\psi\rangle =$ 

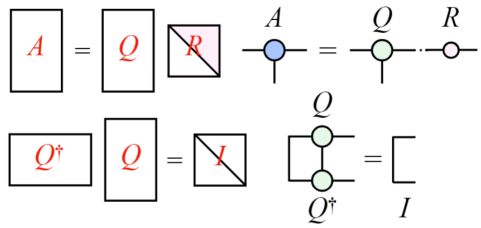
  $= 1$

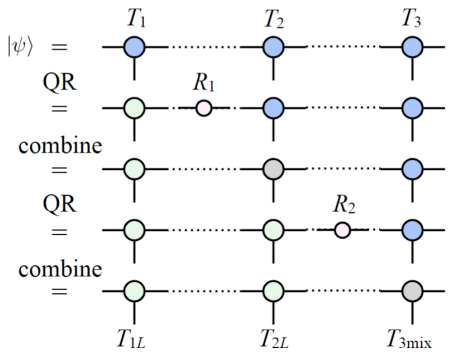
  L: left-canonical; R: right-canonical

  Q: How to find canonical form using QR or LQ?

  - QR decomposition:

    将张量A的 “矩阵化” 做QR分解

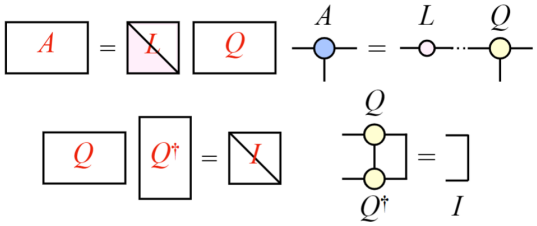
    

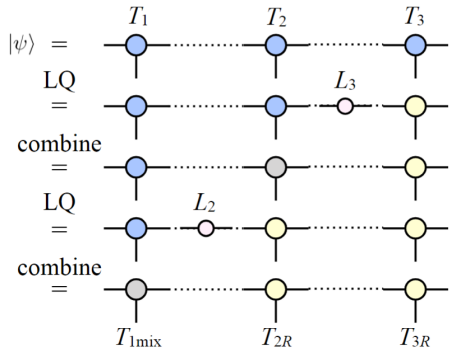
    

    (pushing from left to right)

  - LQ decomposition:

    将张量A的 “矩阵化” 做LQ分解

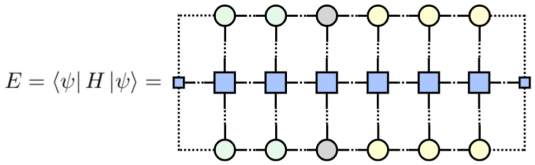




(pushing from right to left)

for periodic boundary condition system: more complicated

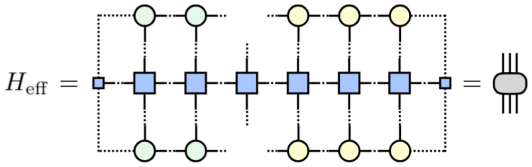
- Variational MPS algorithm (1-site OBC)
  - with open boundary condition and canonical form, we only need to optimize:



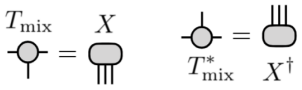
  - idea: we may fix all T_L and T_R and only optimize T_{mix}

    we optimal T_{mix} can be found by solving the **eigenvalue problem**:

$$H_{\text{eff}}\mathbf{x} = E\mathbf{x}.$$



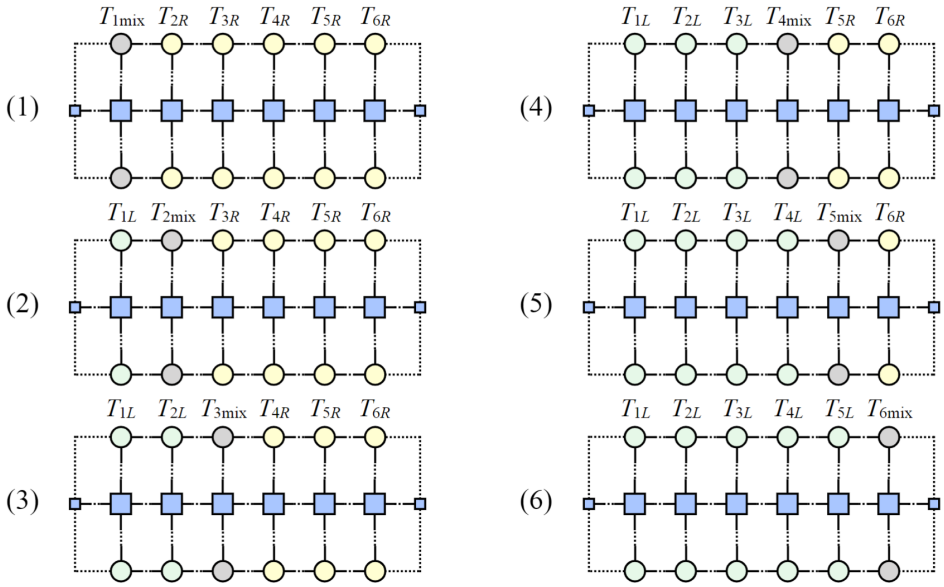
    H_{eff} has 6 open indices in all, we can reshape it into a 2-order tensor (a Hermitian matrix)



    Similarly, T_{mix} can be reshaped into a "vector" $\mathbf{x}$.

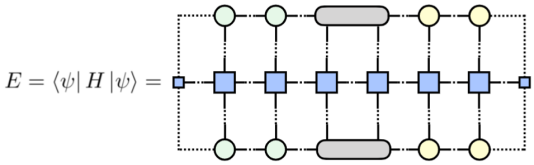
    Minimizing $\mathbf{x}^\dagger H_{\text{eff}}\mathbf{x}$ $\Leftarrow$ diagonalizing H_{eff}.

  - we optimize each site one by one, sweeping back and forth until convergence is reached.

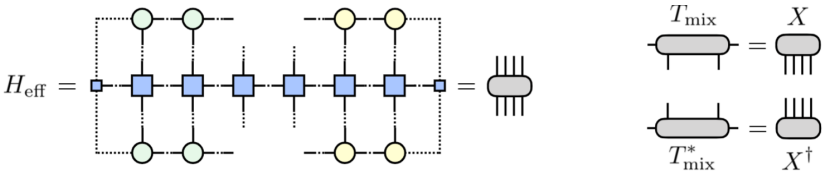


- Variational MPS algorithm (2-site OBC)

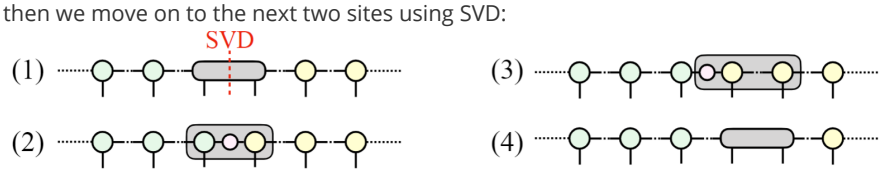
  another option: optimize two sites at once:



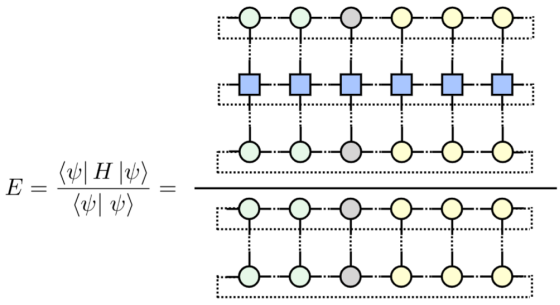
  the optimal T_{mix} can be found, solving the eigenvalue problem: $H_{\text{eff}}\mathbf{x} = E\mathbf{x}$.



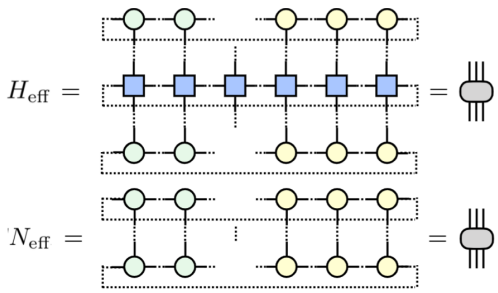
  then we move on to the next two sites using SVD:

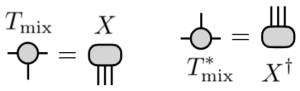


- Variational MPS algorithm (1-site PBC)



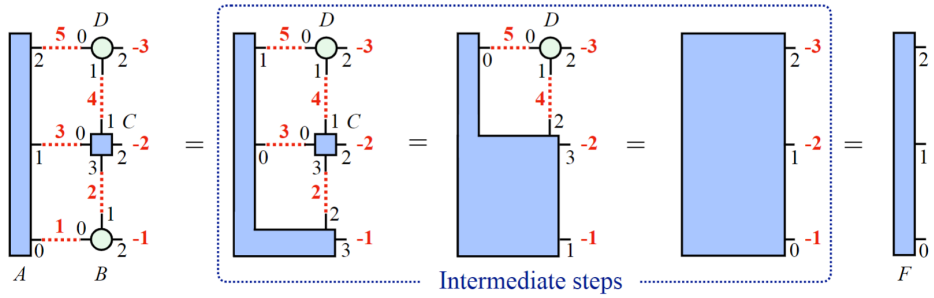
  - because of PBC, $\langle\psi|\psi\rangle \neq 1$, even if we use the canonical form of MPS
  - but we still use the **canonical form for numerical stabilization**
  - we need to solve: a generalized eigenvalue problem $H_{\text{eff}}X = EN_{\text{eff}}X$





T_{mix} corresponds to the ground state of H_{eff} in the presence of N_{eff}

- Subroutine for tensor operations
  - `NCon(Tensors, Indices)` rules:
    - all tensors have their own index order, labeled by $0, 1, 2, \cdots$ in black
    - all indices to be contracted are labeled by $1, 2, 3, \cdots$ in red
    - all indices to be left open are labeled by negative numbers $-1, -2, -3, \cdots$ in red, they become the indices of the final tensor with index order $0, 1, 2, \cdots$ respectively
    - tensors are contracted one by one according to the contraction order $1, 2, 3, \cdots$



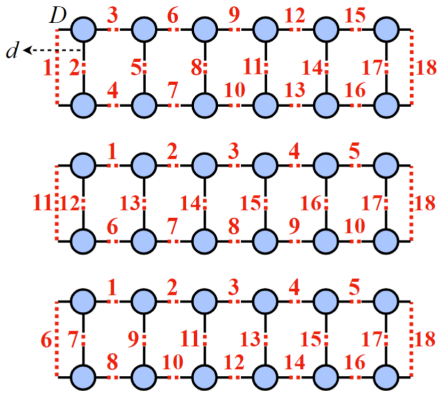
code: $F = \text{NCon}(\,[A, B, C, D],\,[\,[1, 3, 5],\,[1, 2, -1],\,[3, 4, -2, 2],\,[5, 4, -3]\,]\,)$

(0, 1, 2 from A; 0, 1, 2 from B; 0, 1, 2, 3 from C; 0, 1, 2 from D)

R. N. C. Pfeifer, G. Evenbly, S. Singh, G. Vidal, arXiv:1402.0939

more convenient than `numpy.TensorDot` of `numpy.einsum`

  - choice of contraction order: avoiding successive contraction of virtual indices:

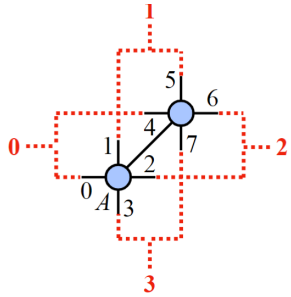


computational cost: $6dD^3 + dD^2$ — best order

computational cost: $2\left(d^2 + d^3 + d^4 + d^5 + d^6\right) D^3 + d^6 D^2$ — worst order

computational cost: $2\left(d^2 + d^3 + d^4 + d^5\right) D^3 + d^6 D^3 + dD^2$

  - group tensor indices to form a new tensor: `Group(A, shape A)`



code: $F = \text{Group}(\,A,\,[\,[0, 4],\,[1, 5],\,[2, 6],\,[3, 7]\,]\,)$

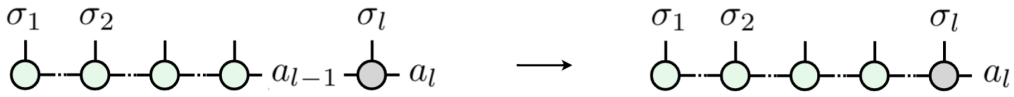
(sub-index from 0; sub-index from 1; sub-index from 2; sub-index from 3)

- equivalence between traditional finite-size DMRG and variational MPS
  - in the traditional DMRG scheme, we grow blocks while decimating basis

$$|a_l\rangle = \sum_{a_{l-1},\sigma_l} \langle a_{l-1}, \sigma_l | a_l \rangle \, |a_l\rangle \, |\sigma_l\rangle.$$

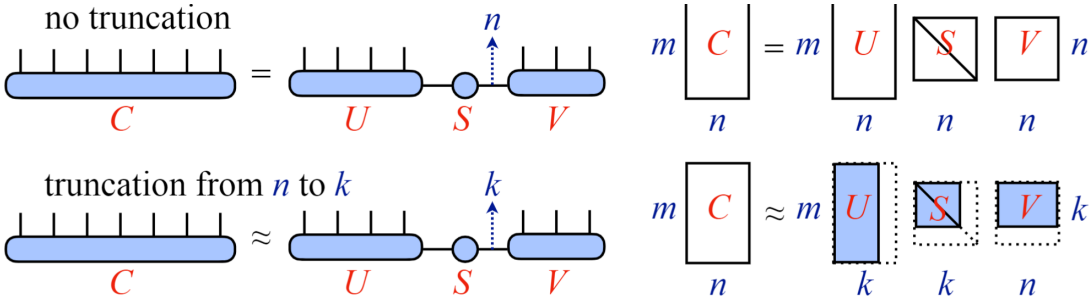
recursion easily expressed as matrix multiplication:

$$|a_l\rangle = \sum_{\sigma_1,\sigma_2,\cdots,\sigma_l} (M^{\sigma_1} M^{\sigma_2} \cdots M^{\sigma_l})_{1,a_l} |\sigma_1, \sigma_2, \cdots, \sigma_l\rangle$$



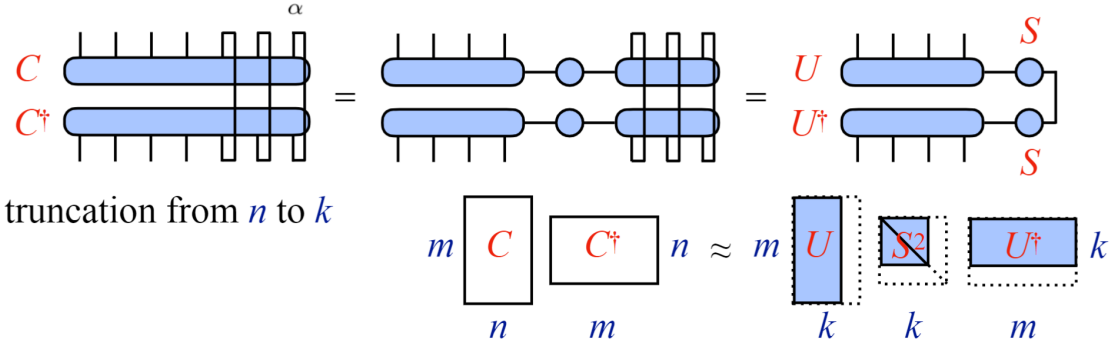
- truncation method: MPS vs DMRG

  MPS

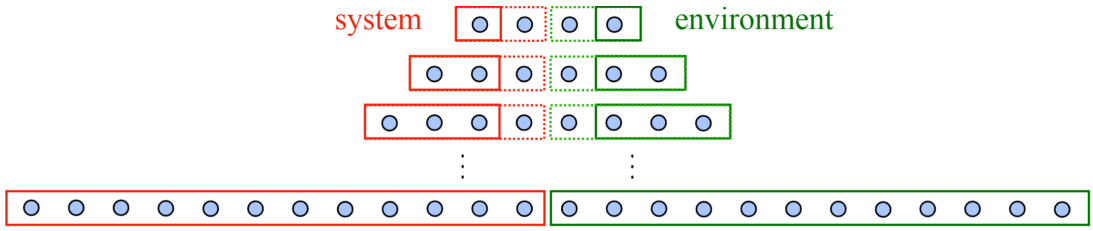


  DMRG

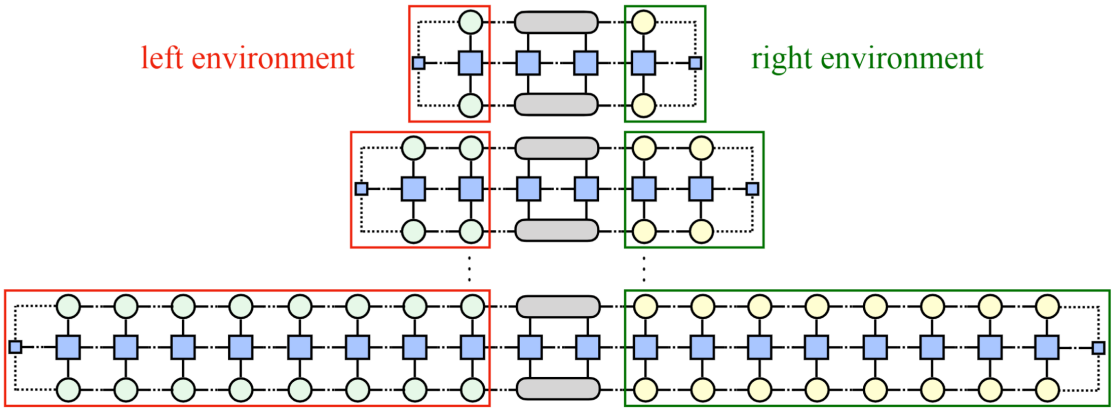
$$\rho_A = \mathrm{Tr}_B \rho = \sum_\alpha s_\alpha^2 |u^\alpha\rangle_{A\,A} \langle u^\alpha| = CC^\dagger = USVV^\dagger SU^\dagger = US^2U^\dagger$$



- iDMRG



  iDMRG based on 2-site update MPS



- iDMRG in the age of MPS (omit)

Infinite Time-Evolving Block Decimation method (iTEBD)

- imaginary-time evolution (ITE):

$$U(\tau) = e^{-H\tau}, \quad |G\rangle = \lim_{\tau\to\infty} e^{-\tau H}|\psi_0\rangle, \quad \text{where} \quad \langle\psi_0|G\rangle \neq 0.$$

- Hamiltonian includes odd terms and even terms (each term involves 2 sites)

$$H = H_{\text{odd}} + H_{\text{even}} = \sum_{i=\text{odd}} h_{i,i+1} + \sum_{i=\text{even}} h_{i,i+1},$$

  $[H_{\text{odd}}, H_{\text{even}}] \neq 0$

  Trotter-Suzuki decomposition

  - first order $e^{(V+W)\delta} = e^{V\delta}e^{W\delta} + \mathcal{O}(\delta^2)$
  - second order $e^{(V+W)\delta} = e^{V\delta/2}e^{W\delta/2}e^{V\delta/2} + \mathcal{O}(\delta^3)$

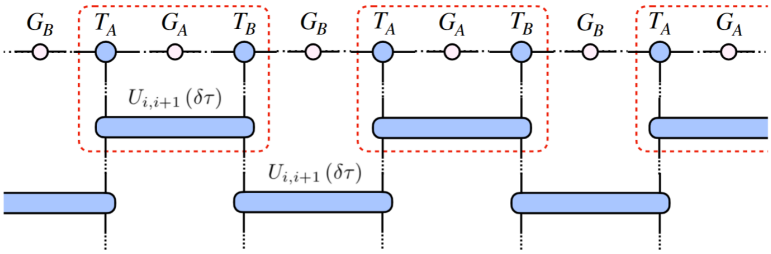
  After Trotter, the evolution operator $U(\tau)$ becomes:

$$U = \prod_{M\to\infty} U(\delta\tau) = \prod_{M\to\infty} \left[\prod_{i=\text{odd}} U_{i,i+1}(\delta\tau)\right]\left[\prod_{i=\text{even}} U_{i,i+1}(\delta\tau)\right]$$

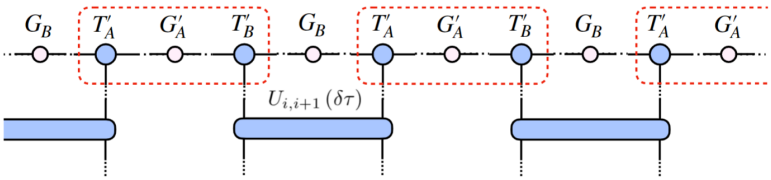
  where

$$U_{i,i+1}(\delta\tau) = e^{-\delta\tau h_{i,i+1}}$$

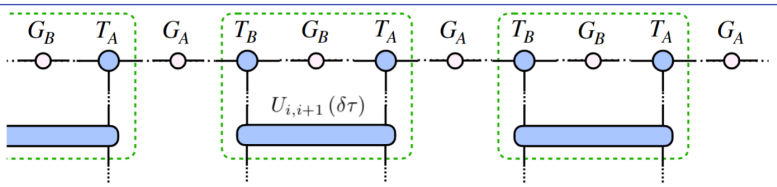
(1)



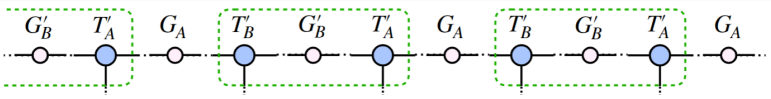
(2)



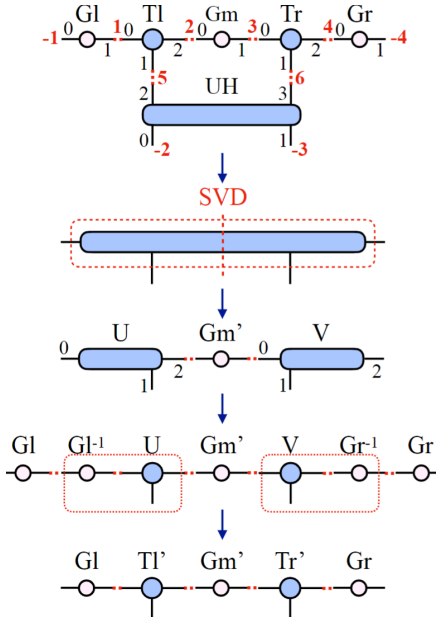
(3)



(4)

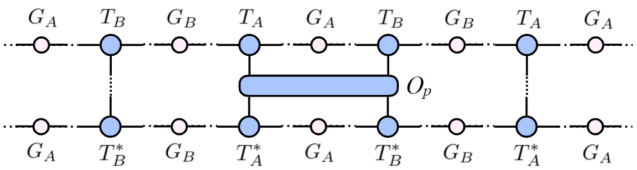


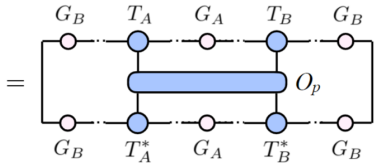
- local update of an MPS

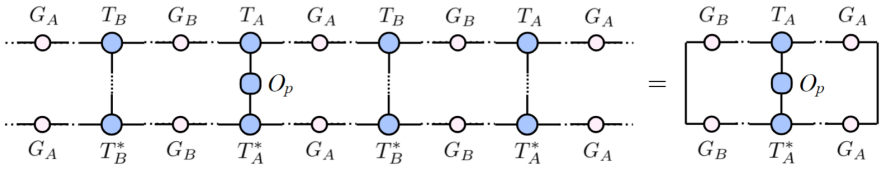


(Tl, Gm, Tr are updated to new values while Gl and Gr remain the same)

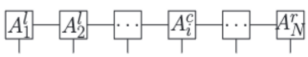
- calculate physical quantities: single- or two- body operator







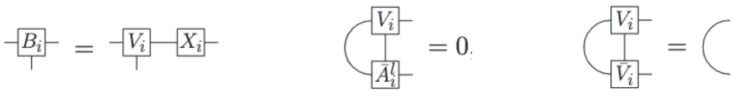
- several applications:
  - Transfer operator and correlation length
  - vMPS for excited states
    - effective Hamiltonian $H^{[1]} = H - E_0 \left|\psi_0\right\rangle\!\left\langle\psi_0\right|$, $H^{[2]} = H - E_0 \left|\psi_0\right\rangle\!\left\langle\psi_0\right| - E_1 \left|\psi_1\right\rangle\!\left\langle\psi_1\right|$
    - quasiparticle ansatz
      - optimal ground-state MPS by standard sweeping algorithm:

$$\left|\Psi(A_1 \dots A_N)\right\rangle =$$ 

      - similar ansatz for excited states:

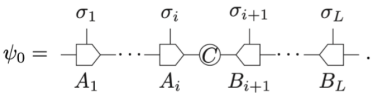
$$\left|\Phi(B_1, \dots, B_N)\right\rangle = \sum_i$$ 

        to ensure the orthogonality condition, B_i should live in the null space of $(A_i^l)^\dagger$, the V_i tensor satisfies the conditions:

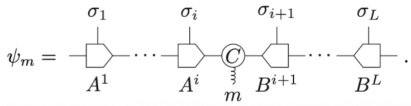


      - excitation ansatz with OBC MPS
      - multi-target MPS (T.Xiang, arXiv:2305.15868)

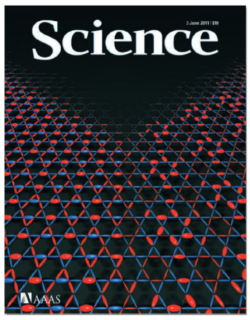
- ground state



- excited states



  - 2D DMRG
    - spin-liquid ground state of the $S = 1/2$ Kagome Heisenberg antiferromagent



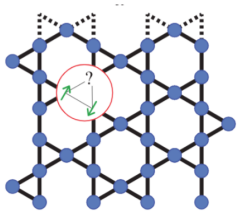
Spin-Liquid Ground State of the $S = 1/2$ Kagome Heisenberg Antiferromagnet

Simeng Yan,[1] David A. Huse,[2,3] Steven R. White[1]*

We use the density matrix renormalization group to perform accurate calculations of the ground state of the nearest-neighbor quantum spin $S = 1/2$ Heisenberg antiferromagnet on the kagome lattice. We study this model on numerous long cylinders with circumferences up to 12 lattice spacings. Through a combination of very-low-energy and small finite-size effects, our results provide strong evidence that, for the infinite two-dimensional system, the ground state of this model is a fully gapped spin liquid.

COVER

Three possible phases of the kagome Heisenberg antiferromagnet, a model of geometrically "frustrated" magnetism: the diamond-pattern valence bond crystal (lower left), honeycomb valence bond crystal (lower right), and quantum spin liquid (upper empty wedge). Deviations of bond strengths from their average values are shown by bond widths and colors (red, stronger; blue, weaker). The simulations of Yan *et al.* (p. 1173) show that the true ground-state phase is the quantum spin liquid.



    - fractional Chern insulators
    - time - dependent DMRG
  - tangent - space method
    - construct the tangent space on the MPS manifold (the space that contains low-energy dynamics)
    - real-time evolution elementary ex

Chap.3 Quantum Monte Carlo

What is Monte Carlo:

- a **very broad** calls of computational algorithms that rely on **repeated random** sampling to obtain numerical results.
- essential idea: use **randomness** to solve problems that might be **deterministic** in principle.

Modern version of Monte Carlo:

- first invented in the late 1940s by *Stanislaw Ulam*, while he was working on nuclear weapons projects at Los Alamos National laboratory. (central to the simulations required for the Manhattan Project)
- (name from the Monte Carlo Casino in Monaco)

Tranditional Monte Carlo

- naive numerical integration using Monte Carlo:

$$\int_a^b f(x)\mathrm{d}x \approx (b-a)\frac{1}{N}\sum_{i=1}^{N} f(x_i) = (b-a)\,\langle f\rangle.$$

  Due to CLT (central limit theorem), the Monte Carlo error decreases with sample size N, as:

$$\varepsilon \propto \frac{1}{\sqrt{N}}$$

- Importance sampling:
  - concentrate the sampling on regions where $f(x)$ is large, using a statistical weight $\omega(x) > 0$ (weight function) with:

$$\int_a^b \omega(x) = 1,$$

the sampling then becomes:

$$\langle f \rangle = \int \frac{f(x)}{\omega(x)} \omega(x) \mathrm{d}x \simeq \frac{1}{N} \sum_{i=1}^{N} \frac{f(x_i)}{\omega(x_i)}.$$

$$\mathrm{Var}\, \langle f \rangle \to \mathrm{Var} \left\langle \frac{f}{\omega} \right\rangle = \frac{1}{n} \mathrm{Var} \frac{f}{\omega}.$$

in order to keep the variance small, the distribution ω should be similar to f

- example of importance sampling:

$$\langle A \rangle = \frac{\sum_n e^{-\beta E_n} A_n}{Z}$$

choose a subset of microstates $\{x_i\}$ with probability distribution:

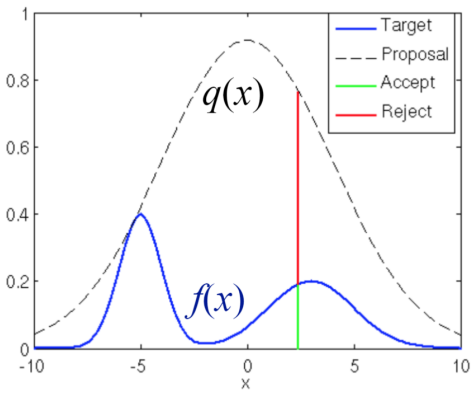
$$\omega(\{x_i\}) = e^{-\beta E(\{x_i\})}$$

then the average becomes:

$$\langle A \rangle = \frac{\frac{1}{N} \sum_{i=1}^{N} \frac{e^{-\beta E(\{x\}_i)}}{\omega(\{x\}_i)} A\left(\{x\}_i\right)}{\frac{1}{N} \sum_{i=1}^{N} \frac{e^{-\beta E(\{x\}_i)}}{\omega(\{x\}_i)}} \simeq \frac{1}{N} \sum_{i=1}^{N} A\left(\{x\}_i\right)$$

the next question is how to generate random variable $\{x_i\}$ with desired probability distribution $\omega(\{x_i\})$?

- Acceptance-rejection sampling
  - suppose: we want to sample from $f(x)$ (which is difficult or impossible to sample from directly), but instead have a simpler distribution $q(x)$ from which sampling is easy.



  - a lot of rejections can take place before a useful sample is generated, thus making the algorithm inefficient and impractical

- Markov chain
  - **Markov process** is a stochastic process (a sequence of random variables $x_0, \cdots, x_n, \cdots,)$

    the probability of moving to the next state depends only on the present state and not on the previous states.

$$P(x_{n+1}|x_0, \cdots, x_n) = P(x_{n+1}|x_n)$$

  - we use a Markov process repeatedly to generate a **Markov chain of states**

    characterization of a Markov chain:

    initial probability distribution $P(x_0)$, and the transition probability $W(x_n \to x_{n+1})$

  - **equilibrium condition**:

$$\sum_y P(x) W(x \to y) = \sum_y P(y) W(y \to x)$$

    conditions to satisfy:

    1. $\sum_y W(x \to y) = 1$, therefore the equilibrium condition is simplify to:

$$P(x) = \sum_y P(y) W(y \to x).$$

2. ergodicity: it should be possible to reach any state from any other state, if we run it for long enough
3. detailed balance:

$$P(x)W(x \to y) = P(y)W(y \to x)$$

on average the system should go from x to y just as often as it goes from y to x

- Metropolis-Hasting algorithm
  - two parts of transition probability:

$$W(x \to y) = Q(x \to y)A(x \to y),$$

$Q(x \to y)$ is **selection probability**: $Q(x \to y) = P(y|x)$

$A(x \to y)$ is **acceptance probability**: $A(x \to y) \in (0, 1)$ (acceptance ratio), if we start from x and our algorithm generates a new state y from it, we can accept or reject state y with

  - for each t
    - randomly generate a candidate state y according to $Q(x \to y)$
    - calculate the acceptance probability:

$$A(x, y) = \min\left\{1, \frac{P(y)}{P(x)}\right\}(\text{Metropolis}) = \min\left\{1, \frac{P(y)Q(y \to x)}{P(x)Q(x \to y)}\right\}(\text{Metropolis-Hasting})$$

    - generate a u from $U(0, 1)$

      if $u \leq A(x, y)$, accept the new state and set $x_{t+1} = y$

      if $u > A(x, y)$, reject the new state and copy the old state forward $x_{t+1} = x$.
    - we can easily verify that the choice of Q and A ensures the **detailed balance condition**
- Example: Monte-Carlo simulation of classical Ising model
  - choose an initial configuration for the spins
  - we propose a move by randomly choosing a spin and flip it
  - **acceptation**
    - $\sigma_j \to -\sigma_j$, flip the j-th spin
    - evaluate:

$$\alpha = \exp(-\beta\Delta E), \quad \Delta E = 2\sigma_j\left(J\sum\sigma_{\text{near}} - H\right)$$

    - then we generate a uniform random number $r \in (0, 1)$

      $r < \alpha$, accept the move, $\{\sigma_{\text{new}}\} = \{\sigma_{\text{trial}}\}$

      $r > \alpha$, reject the move, $\{\sigma_{\text{new}}\}$ remains the same
  - iteration: use new configuration as the new starting point and go back to the trial step
  - equilibration: we discard the first steps of the random walk, when the distribution of the sampled Markov chain has not yet reached its limit $P(x)$
  - data blocking

Quantum Monte Carlo - world line QMC

- $\langle A\rangle = \frac{1}{Z}\text{Tr}\{Ae^{-\beta H}\}$ thermal expectation value

  where $\beta = 1/T$ and possibly $T \to 0$
- $Z = \text{Tr}\{e^{-\beta H}\} = \text{Tr}\{\prod_{l=1}^{L} e^{-\Delta_\tau H}\}$, $\Delta_\tau = \beta/L$ (time slicing of the partition function)

  choose a basis and insert complete sets of states:

$$Z = \sum_{\alpha_0,\cdots,\alpha_{L-1}} \langle\alpha_0|e^{-\Delta_\tau H}|\alpha_{L-1}\rangle \cdots \langle\alpha_2|e^{-\Delta_\tau H}|\alpha_1\rangle \langle\alpha_1|e^{-\Delta_\tau H}|\alpha_0\rangle$$

  use approximation for imaginary time evolution operator

$$Z \approx \sum_{\{\alpha\}} \langle\alpha_0|1-\Delta_\tau H|\alpha_{L-1}\rangle \cdots \langle\alpha_2|1-\Delta_\tau H|\alpha_1\rangle \langle\alpha_1|1-\Delta_\tau H|\alpha_0\rangle$$

  leads to error $\propto \Delta_\tau$

- we can compute the weights for the different time periodic paths

$$\alpha_0 \to \alpha_1 \to \cdots \to \alpha_{L-1} \to \alpha_0,$$

these paths are importance sampled according to their weight

- After detailed equilibrium is reached, the expectation value of the operator O is:

$$\langle O \rangle = \frac{1}{Z}\mathrm{Tr}\left(Oe^{-\beta H}\right) = \frac{1}{Z}\sum_{\{\alpha\}} \langle \alpha_0 | e^{-\Delta_\tau H} | \alpha_{L-1} \rangle \cdots \langle \alpha_2 | e^{-\Delta_\tau H} | \alpha_1 \rangle \langle \alpha_1 | e^{-\Delta_\tau H} O | \alpha_0 \rangle.$$

write this in a form suitable for MC importance sampling:

$$\langle O \rangle = \frac{\sum_{\{\alpha\}} O(\{\alpha\}) W(\{\alpha\})}{\sum_{\{\alpha\}} W(\{\alpha\})},$$

W — weight, O — estimator

Quantum Monte Carlo - stochastic series expansion (SSE)

- Taylor expansion of the Boltzmann operator:

$$e^{-\beta H} = \sum_{n=0}^{\infty} \frac{(-\beta)^n}{n!} H^n,$$

choosing a basis, the partition function can be written as:

$$Z = \sum_{n=0}^{\infty} \frac{(-\beta)^n}{n!} \sum_{\{\alpha\}_n} \langle \alpha_0 | H | \alpha_{n-1} \rangle \cdots \langle \alpha_2 | H | \alpha_1 \rangle \langle \alpha_1 | H | \alpha_0 \rangle,$$

for any model, the energy is:

$$\begin{aligned} E &= \frac{1}{Z}\sum_{n=0}^{\infty} \frac{(-\beta)^n}{n!} \sum_{\{\alpha\}_{n+1}} \langle \alpha_0 | H | \alpha_{n-1} \rangle \cdots \langle \alpha_2 | H | \alpha_1 \rangle \langle \alpha_1 | H | \alpha_0 \rangle \\ &= -\frac{1}{Z}\sum_{n=1}^{\infty} \frac{(-\beta)^n}{n!} \cdot \frac{n}{\beta} \sum_{\{\alpha\}_n} \langle \alpha_0 | H | \alpha_{n-1} \rangle \cdots \langle \alpha_2 | H | \alpha_1 \rangle \langle \alpha_1 | H | \alpha_0 \rangle \\ &= -\frac{\langle n \rangle}{\beta}, \end{aligned}$$

- **fixed length scheme**: cut-off at $n = L$, fill in with unit operators I:

$$Z = \sum_S \frac{(-\beta)^n (L-n)!}{L!} \sum_{\{\alpha\}_L} \sum_{\{S_i\}} \langle \alpha_0 | S_L | \alpha_{L-1} \rangle \cdots \langle \alpha_2 | S_2 | \alpha_1 \rangle \langle \alpha_1 | S_1 | \alpha_0 \rangle.$$

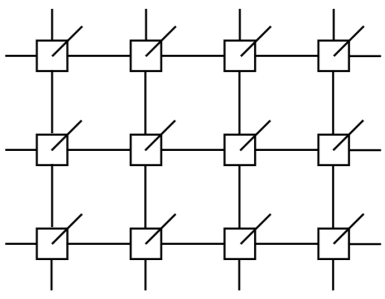
here, n is the number of $S_i = H$ instances in the sequence S

- frustrated systems have sign problems
- updating process

Chap.4 Tensor Network State

Outline: Projected Entangled-Pair States (PEPS, with application in topological order), 2D tensor network algorithms, Multi-scale Entanglement Renormalization Ansatz (多尺度纠缠重整化，MERA), etc.

PEPS



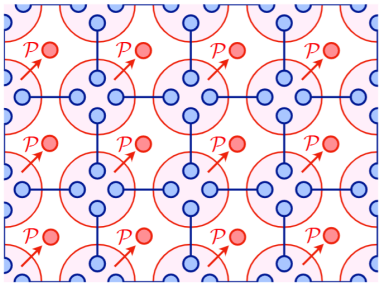
- representation of some simple states:

- product state

$$|\psi\rangle = (\alpha|0\rangle + \beta|1\rangle)^{\otimes N}$$

$$\mathcal{P} = \alpha|0\rangle\left\langle \begin{matrix} & 0 & \\ 0 & & 0 \\ & 0 & \end{matrix}\right| + \beta|1\rangle\left\langle \begin{matrix} & 0 & \\ 0 & & 0 \\ & 0 & \end{matrix}\right|, \qquad |\omega\rangle = |0,0\rangle$$

"project out virtual freedoms, reconstruct physical freedoms":



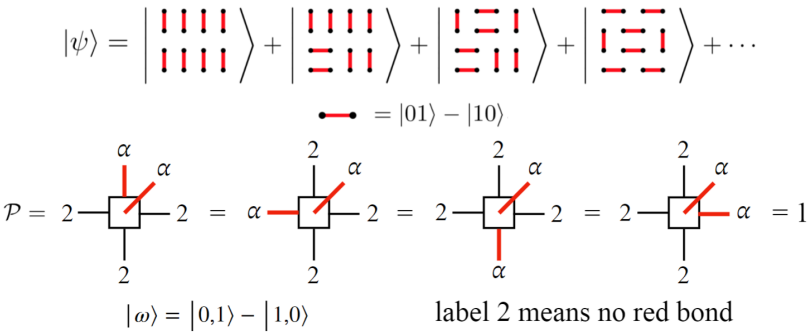
product state needs only 1 virtual freedoms and 2 physical freedoms

- GHZ state $|\psi\rangle = |0\cdots0\rangle + |1\cdots1\rangle$

$$\mathcal{P} = |0\rangle\left\langle \begin{matrix} & 0 & \\ 0 & & 0 \\ & 0 & \end{matrix}\right| + |1\rangle\left\langle \begin{matrix} & 1 & \\ 1 & & 1 \\ & 1 & \end{matrix}\right|, \qquad |\omega\rangle = |0,0\rangle + |1,1\rangle$$

product state needs 2 virtual freedoms

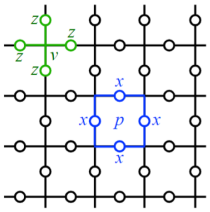
- RVB state (resonated valence bond state)



virtual freedoms: 3

- Kitaev's toric code: the simplest and most well studied spin model with topological order

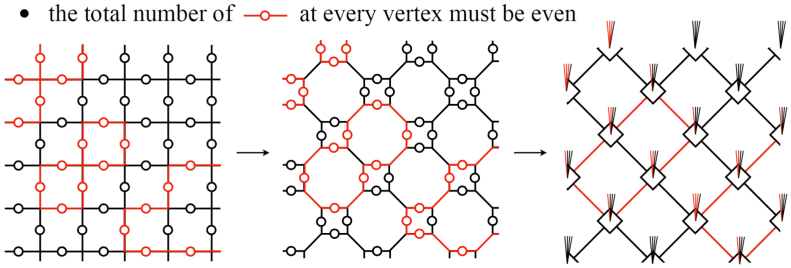
$$H = -\sum_{v} \prod_{j \in \text{Vertex}} Z_j - \sum_{p} \prod_{j \in \text{Plaquatte}} X_j,$$



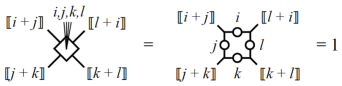
ground states are **equal weight superposition** of **ALL CLOSED LOOPS**

PEPS representation?

- the total number of  at every vertex must be even

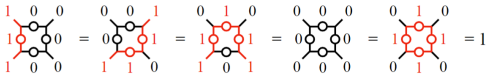


- non-trivial elements of PEPS



$$i,j,k,l \in \{0,1\}$$
$$[\![i+j]\!] = \mathrm{mod}\,(i+j,2)$$

- example

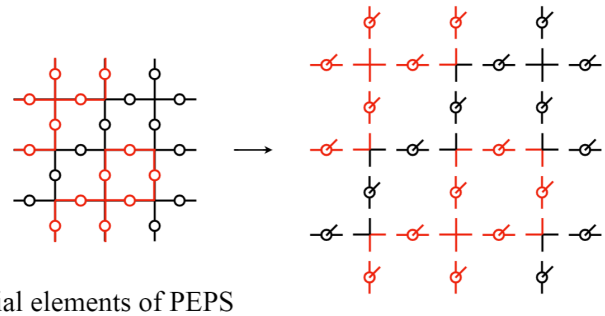


physical freedoms: 16, virtual freedoms: 2 (red or black)

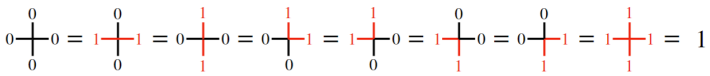
non-trivial elements: satisfying $[\![i+j]\!] = \mathrm{mod}(i+j,2)$

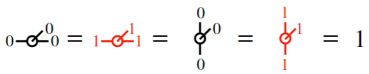
Another representation:

- the total number of at every vertex must be even



- non-trivial elements of PEPS

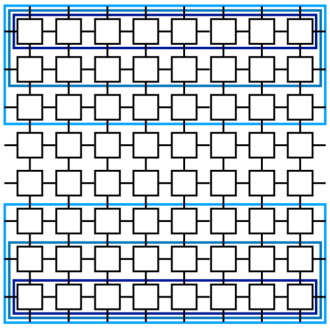
vertex tensor 

bond tensor 

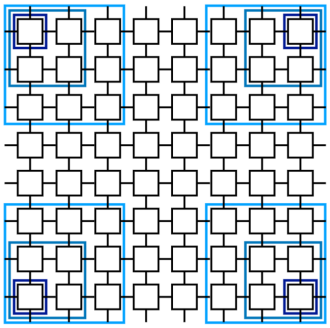
2D Tensor Network

- exact contraction of PEPS is a #P hard problem

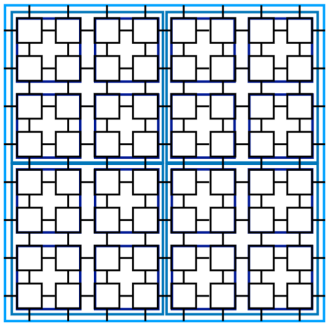
approximation methods to contract 2D tensor networks:



boundary MPS



corner transfer matrices



tensor coarse-graining