TEBD (time evolving block decimation) algorithm

Goal: time evolution $|\psi(t)\rangle = e^{-iHt}|\psi(t=0)\rangle$

Consider nearest-neighbor Hamiltonian $H = \sum_i h_{i,i+1} = \sum_{i \text{ even}} h_{i,i+1} + \sum_{i \text{ odd}} h_{i,i+1} = H^e + H^o$ (only acts on sites i, i+1)

Suzuki Trotter decomposition:

loop over many small time steps δ until we reach t

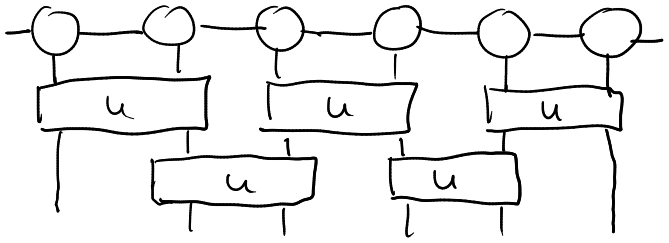
$$e^{-iH\delta} = e^{-iH^e\delta} e^{-iH^o\delta} + O(\delta^2)$$

or second order $$e^{-iH\delta} = e^{-iH^e\frac{\delta}{2}} e^{-iH^o\delta} e^{-iH^e\frac{\delta}{2}} + O(\delta^3)$$

(higher order formulas available)

with exact exponentials $e^{-iH^e\delta} = e^{-i(\sum_i h_{i,i+1})\delta} = \prod_{i \text{ even}} e^{-i h_{i,i+1}\delta}$, $e^{-i h_{i,i+1}\delta} = U$

$e^{-iH\delta}|\psi\rangle =$ $|\psi\rangle$, $e^{-iH_e\delta}$, $e^{-iH_o\delta}$

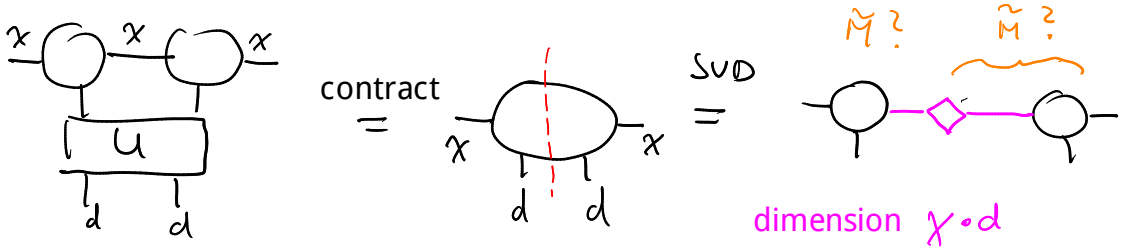


"Brick wall"

alternative: left-right sweep "staircase" for imaginary time evolution

How to get back into MPS form?

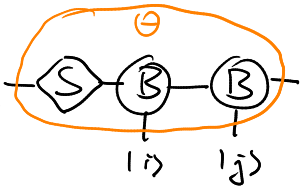
consider applying a single 2-site gate naively:



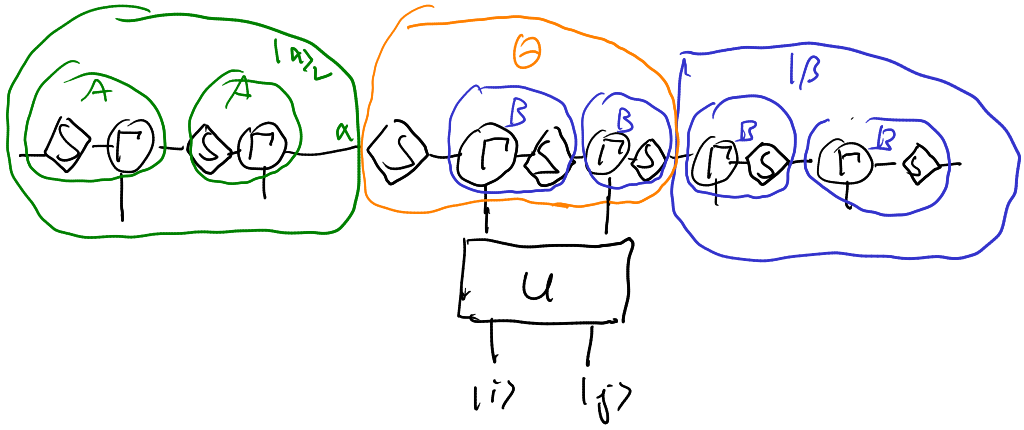
Problem: bond dimension at center bond grows, we need to truncate $\chi \cdot d \to \chi$

To get a globally optimal truncation, we need an ONB!

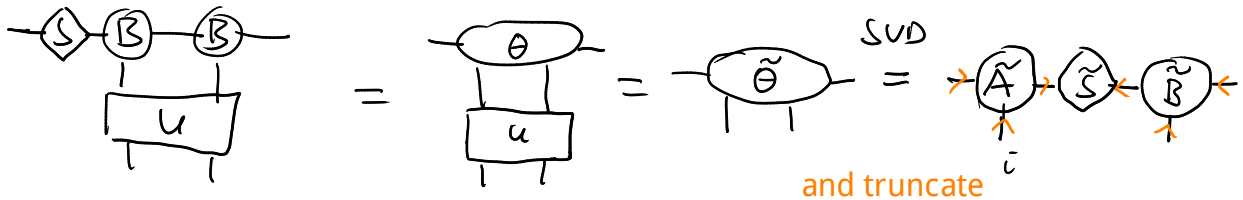
Trick: save S and right-canonical B on each site (also good for local expectation values)

then $|\Theta\rangle = |\alpha\rangle_L$  $|\beta\rangle_R$ is local representation of ψ

in ONB $|\alpha\rangle_L\ |i\rangle\ |j\rangle\ |\beta\rangle_R$

since $U|\psi\rangle =$ 

Thus, the *correct* TEBD update is:

$U|\Theta\rangle =$ 

and truncate

Then use ⊛ to get left tensor back into B form:

left Schmidt values don't change for local unitary U within one side
but if truncation becomes too large, Schmidt values are no longer correct!

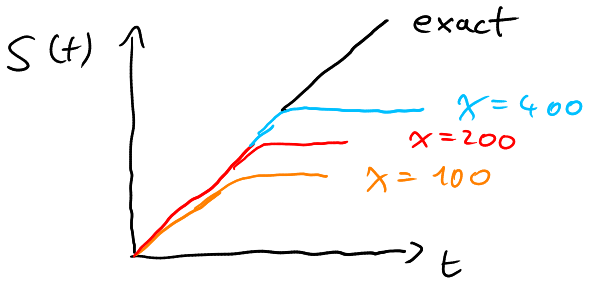
Note: we can trivially apply the U on all odd (even) sites in parallel

Note: the very same update scheme also works with infinite MPS, see below

Sources of errors: - Trotter error => choose small time step

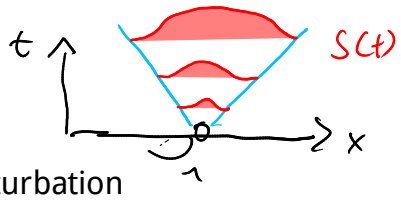
- truncation error => allow to increase bond dimension over time

for a global quench: typically $S(t) \sim t$ so we would need $\chi(t) \sim e^t$



=> we need to stop time evolution
when truncation errors become too large!

for local quench:



light-cone
(Lieb Robinson bounds)

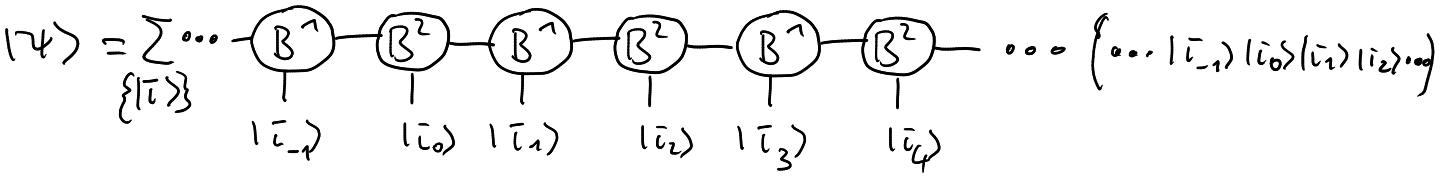
often tractable up to t= 20... 50
e.g. for calculations of spectral function

$$\langle \hat{X}(t)\, \hat{Y} \rangle = \langle \psi_0 | e^{iE_0 t} \hat{X}\, \underbrace{e^{iHt} Y |\psi_0\rangle}_{|\psi(t)\rangle}$$

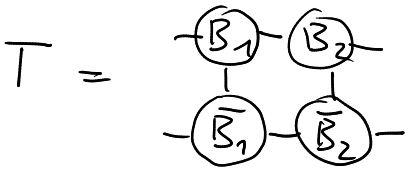
Infinite MPS / UMPS (Uniform MPS)

H is translation invariant, but we only need local updates
so we can directly work in the thermodynamic limit $N \to \infty$

Instead of having different tensors on each site, consider a unit cell of length L, e.g. L=2



left and right infinite parts in overlaps / expectation values
collapse to dominant eigenvectors of the transfer matrix



local updates of the tensors within the unit cell
effectively act in parallel on all unit cells

Strictly, the local truncation is not globally optimal anymore

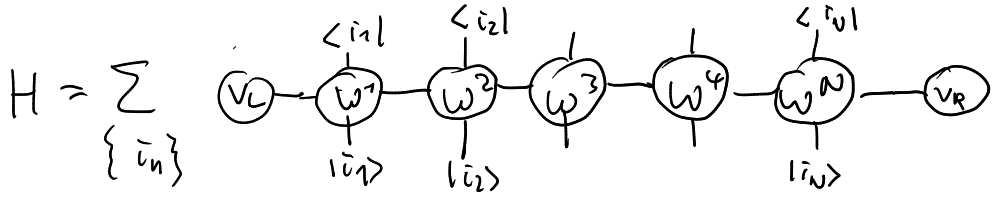
-> one can do mathematically rigurous time dependent variational principle (TDVP)
in the tangent space of MPS

see Vanderstaten et al. arXiv:1810.07006

Matrix Product Operator (MPO)

= generalization of MPS to operators

conventient for Hamiltonians beyond nearest neighbors



translation invariant Hamiltonian: same W everywhere

Examples

$$W_i = \begin{pmatrix} \mathbb{1}_i & Z_i & -gX_i \\ & & -JZ_i \\ & & \mathbb{1}_i \end{pmatrix}, \quad v_L = (1\ 0\ 0), \quad v_R = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

matrix=bonds
entries=local operators

$$\rightsquigarrow v_L W_1 W_2 v_R = (\mathbb{1}_1, Z_1, -gX_1) \begin{pmatrix} -gX_2 \\ -JZ_2 \\ \mathbb{1}_2 \end{pmatrix}$$

$$= -g\,\mathbb{1}_1 X_2 - J Z_1 Z_2 - g X_1 \mathbb{1}_2$$

general N: $H = -J \sum_{i<N} Z_i Z_{i+1} - g \sum_i X_i$ ✓

general: upper-triangular block structure

$$W = \begin{pmatrix} \mathbb{1} & C_n & D_n \\ 0 & A & B \\ 0 & 0 & \mathbb{1} \end{pmatrix}$$

D = onsite term
C B = nearest-neighbor torm
A = long-range structure

$$\rightsquigarrow H = \sum_n D_n + \sum_n C_n B_{n+1} + \sum_n C_n A B_n + \sum_n C_n A_{n+1} A_{n+2} B_{n+3}$$

⇝ one can formalize construction of W to auto-generate MPOs for many Hamiltonians

note: W not diagonalizeable! Jordan structure

$$H = \begin{pmatrix} 1 & \varepsilon \\ & 1 \end{pmatrix} \rightsquigarrow \begin{pmatrix} 1 & \varepsilon \\ & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} + \varepsilon \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

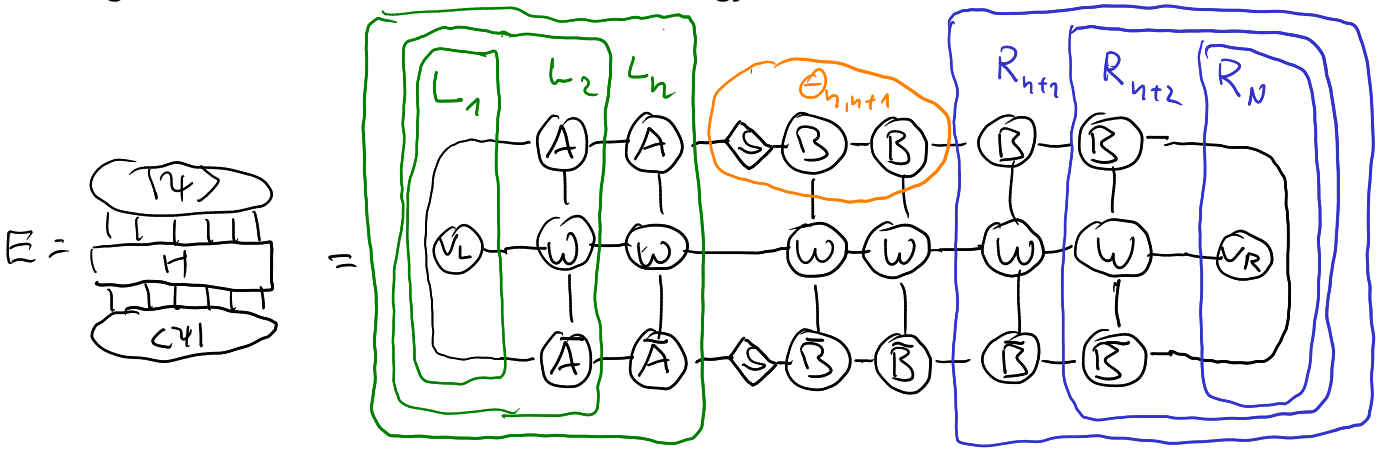
$$\begin{pmatrix} 1 & \varepsilon \\ & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

energy density per site

H is extensive,
applying transfer matrix
adds energy density

Density Matrix Renormalization Group (DMRG) algorithm

goal: variational minimization of the energy $E = \langle \psi | H | \psi \rangle$



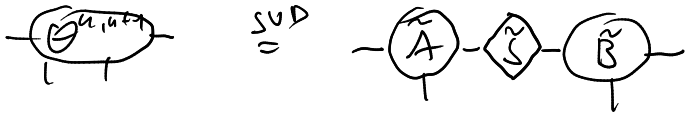
recursively define left/right environments L/R
which contain only left/right-canonical MPS tensors (and MPO)

Finite DMRG: sweep algorithm

Start with an MPS in right canonical form

for n in {1, 2, ... N-1}: right sweep

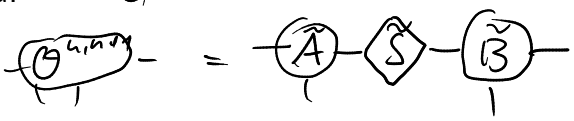
find the optimal  keeping other tensors fix

split  and truncate

use $\tilde{A}$ to calculate next left environment L^{n+1}

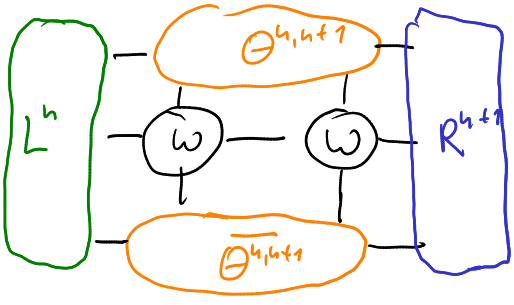
for n in {N-2, ..., 1}: left sweep

find optimal $\Theta^{n,n+1}$

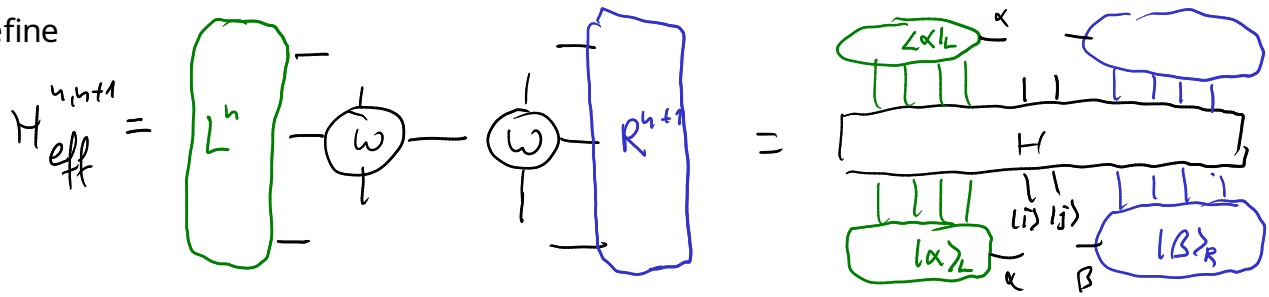
split  and truncate

use $\tilde{B}$ to calculate next right environment R^n

How to find optimal $\Theta^{n,n+1}$?

$E =$ 

define

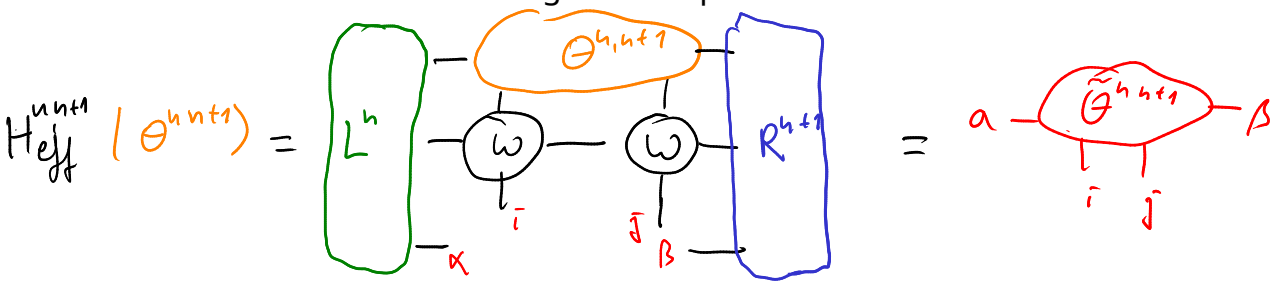


This is the Hamiltonian projected into the basis $|\alpha\rangle_L \otimes |i\rangle \otimes |j\rangle \otimes |\beta\rangle_R$

This is an orthonormal basis!

Hence, the optimal  is simply given by diagonalizing $H_{eff}^{n,n+1}$

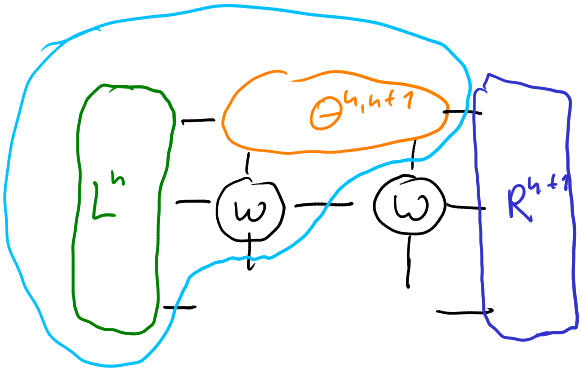
viewed as a matrix acting on the top indices



The ground state of $H_{eff}^{n,n+1}$ can be found with Lanczos algorithm (or similar)

Note: contraction order matters!

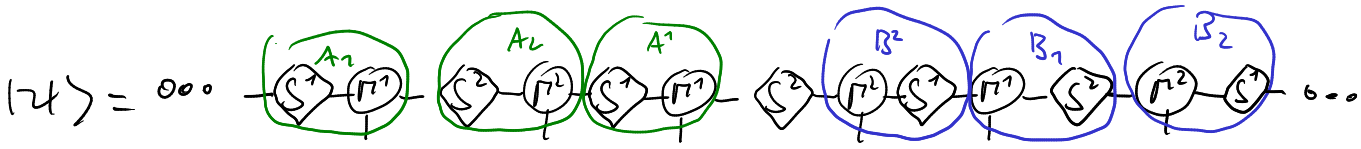
$$\mathcal{O}(\chi^3 d^2 D)$$



Infinite DMRG

consider a system which is translation invariant with unit cells with N sites

here: N=2



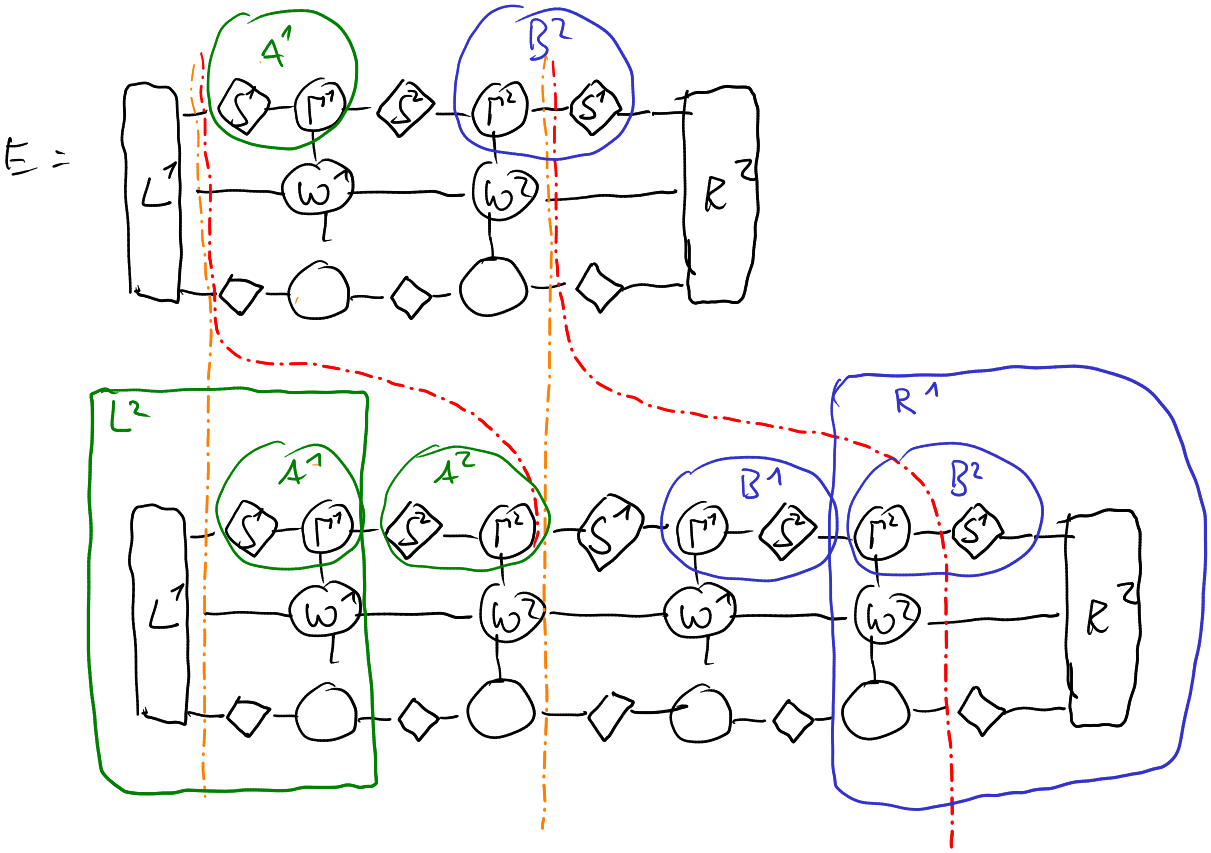
Start with a small system of just one unit cell

sweep left-right left in a single unit cell (as in finite DMRG)
but include the bond between unit cells (which is trivial for finite DMRG)

During each sweep, grow the left/right environets by one unit cell,
assuming translation invariance

Repeat until actual translation invariance is reached

Example: N=2



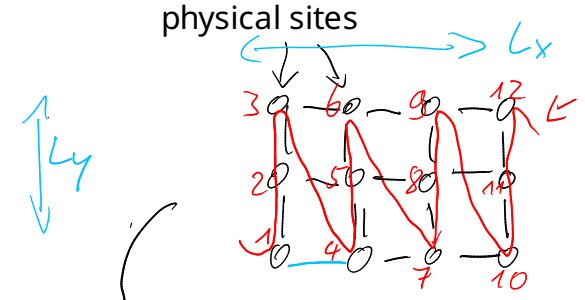
Note: total energy increases by energy density!

towards 2D: DMRG on a cylinder

MPS = 1D ansatz, but: MPO allows long-range hopping

idea: map 2D system to a 1D chain

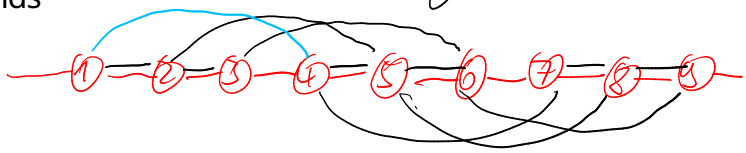
physical sites



MPS "snake"

pull on MPS ends

long range hopping!



price: area law predicts $S = \text{const} \cdot L_y \leq \log \chi$

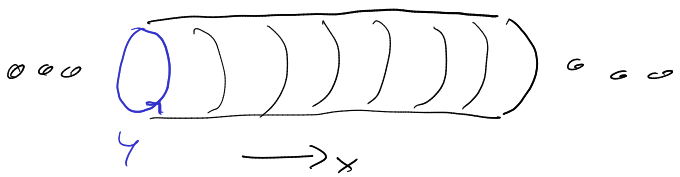
$\Leftrightarrow \chi \geq \exp(\text{const} \cdot L_y)$

$\Rightarrow$ L_y needs to be small, but L_x can be big

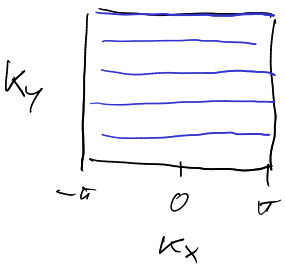
need to study scaling/dependence on L_y, often severe finite-size effects!

periodic boundary conditions in y-direction often reduce finite-width effect

$\rightsquigarrow$ infinite, narrow cylinder



allowed momenta in Brilluioun zone: Ly cuts



can shift cuts by
"threading flux through the cylinder"
= particles hopping around the cylinder
pick up a phase.