First obstacle: Observables (I.1)

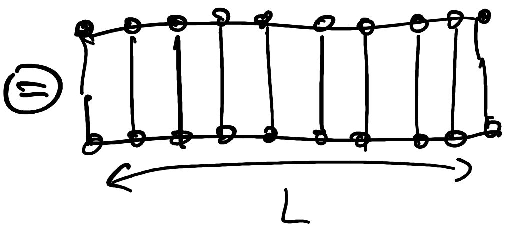
For 1D TN's (i.e. MPS) observables can be evaluated exactly at polynomial costs. For 2D TN's this no longer true and in the absence of extra structure:

> Exact contraction of 2D TN's is Exponentially hard

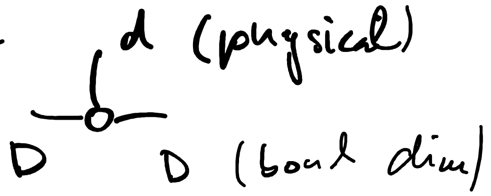
Let's observe difference between finite-size MPS and finite PEPS:

Cost to evaluate a norm (assuming the MPS is not given in the canonical form)

$\langle MPS | MPS \rangle =$



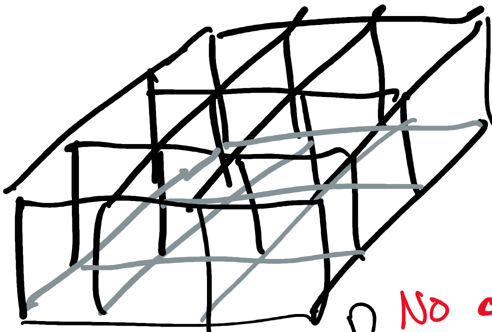
where tensor dimensions are



# of floating point ops is $\sim L D^3 d$

i.e. linear in size & cubic in D.

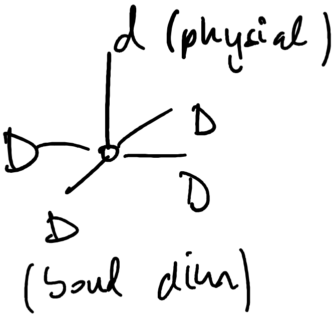
Now finite PEPS:

$\langle PEPS | PEPS \rangle =$ 

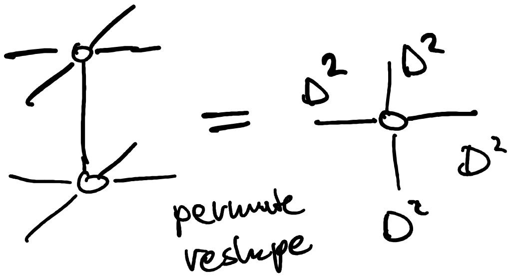
No canonical form, this cannot be simplified

Looking at this TN from above:

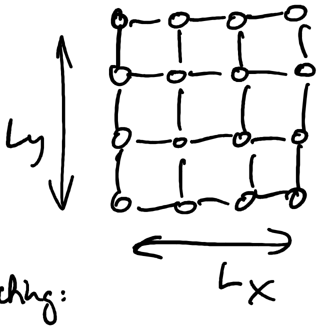
on-site tensor



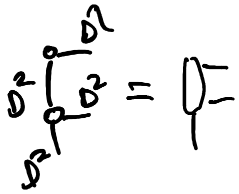
double-layer tensor by contracting over physical index



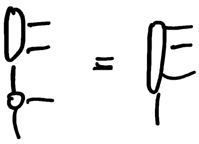
define

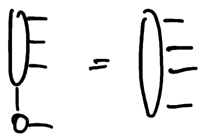
$\langle PEPS | PEPS \rangle =$ 

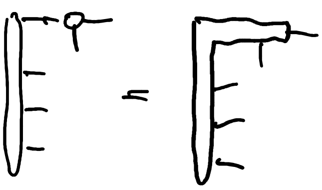
Let's start contracting:

i)  #ops : $(D^2)^4$

(Product of dimensions of outgoing indices and contracted indices)

ii)  #ops : $(D^2)^5$

iii)  #ops : $(D^2)^5$

iv)  #ops: $(D^2)^6$

& continue ...

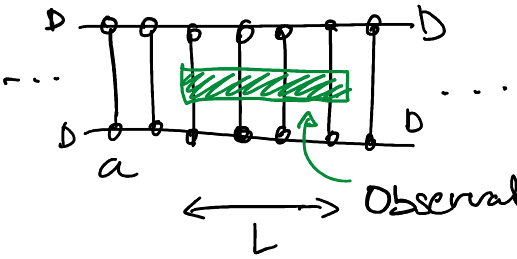
Total number of operations

$$\propto \left(D^2\right)^{\min(L_x, L_y)+2}$$

this is Exponential(!) in width / height of the system.

* What about infinite MPS (iMPS)?

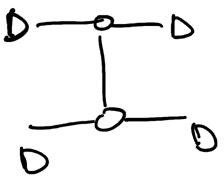
generated by single tensor $\underline{a}$ (i.e. 1site translationary invariant iMPS)



$= \langle iMPS | \hat{O} | iMPS \rangle$

Observable supported on L-site interval

The boundaries of any interval are given by leading left & right eigenvectors of transfer operator



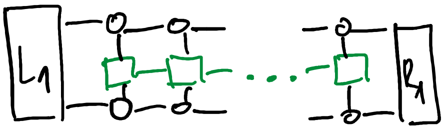
$= \mathbb{T} \in$ Matrix $D^2 \times D^2 = \sum_{i=1}^{D^2} \boxed{R_i} \, \lambda_i \, \boxed{L_i}$

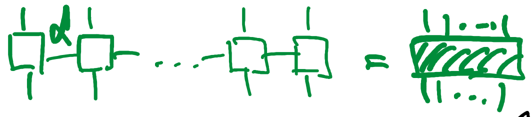
injectivity of iMPS will guarantee $\lambda_1 > \lambda_{i>1}$

i.e. unique leading eigenvalue & eigenvectors

The cost of diagonalizing $\mathbb{T}$ is $O((D^2)^3)$ or to get just a few leading eigenpairs $O((D^2)^4)$.

With the knowledge of $\boxed{R_1}$, $\boxed{L_1}$ ANY observable on the interval of length L can be evaluted EXACTLY by contracting TN:



where 

i.e. MPO corresponding to $\hat{O}$.
with bond dimension α.

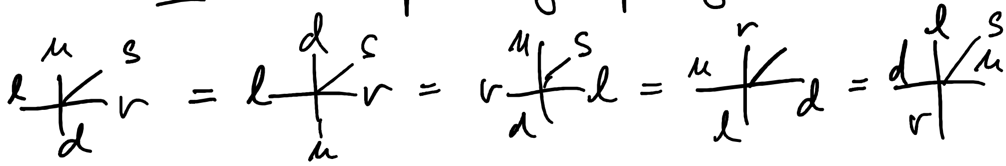
The total cost of contraction is

$$\propto L D^3 \alpha d$$

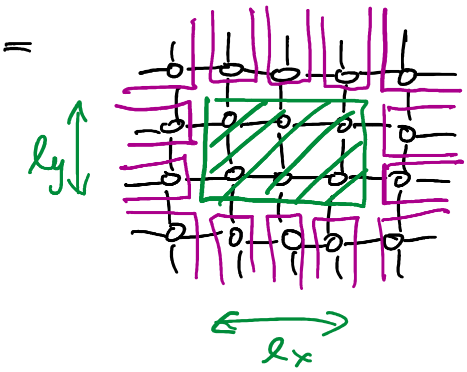
hence the diagonalization of Π $O(D^4)$ dominates.

Q: Infinite PEPS (iPEPS): How to move forward? (I.2)

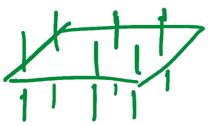
We work with 1-site translationary invariant iPEPS generated by tensor $\underline{a}$ with point-group symmetries

$a^{s}_{\mu l d r} =$ 

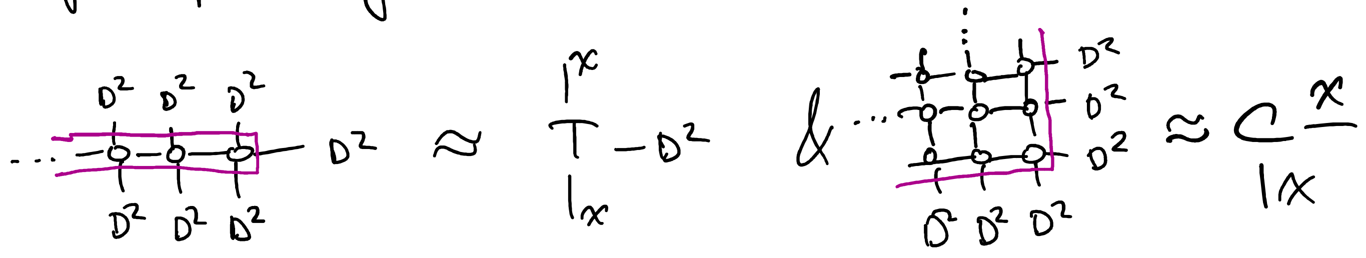
We consider only observables with finite support, i.e. ones defined on a finite $l_x \times l_y$ patch. Typical examples are: interaction terms of local Hamiltonian, local observables, correlation functions, ...

$\langle iPEPS | \hat{O} | iPEPS \rangle =$ 

(a top view of double-layer TN)

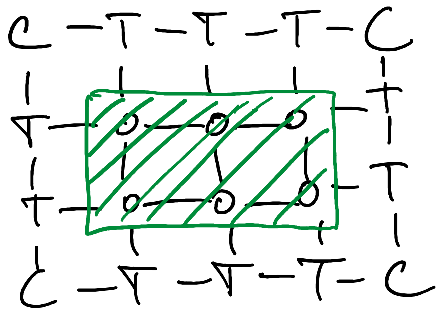
 $= \hat{O}$

ASSUME there exists C matrix of size $\chi \times \chi$ and T tensor of size $\chi \times \chi \times D^2$, which approximate infinite parts of the above TN as



Then, the expectation value of $\hat{O}$ can be approximated through C, T as

$$\langle \text{iPEPS} | \hat{O} | \text{iPEPS} \rangle \approx$$



$$= \langle \hat{O} \rangle_\chi$$

The error of the approximation can be controlled by increasing χ, with $\chi \to \infty$ limit recovering exact values

$$\lim_{\chi \to \infty} \langle \hat{O} \rangle_\chi = \langle \text{iPEPS} | \hat{O} | \text{iPEPS} \rangle$$

The contraction of approximate network $\langle \hat{O} \rangle_\chi$ is still exponential in minimal linear size of the observable $\min(l_x, l_y)$. But for many practical use cases, i.e.

* Nearest neighbour interaction, next-nearest-neighbour interaction, ...

* local magnetization, VBS order parameter
* 2-point correlation functions in vertical or horizontal directions

the computational cost is not prohibitive, in particular if convergence with χ is fast. [generally the required χ is proportional to correlation length.]

Why should such C, T exist and how to find them (I.3)

$\boxed{\text{TL;DR}}$ Summary

To perform approximate contraction of infinite TN corresponding to $\langle \text{iPEPS} | \text{iPEPS} \rangle_{t=D^2}$ or $Z_{t=D}$ of classical 2dim. stat-mech model

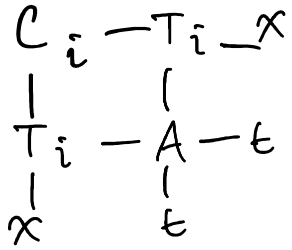
i) pick initial C_0, T_0.

*) random initialization is also possible

*) C_0, T_0 can be embedded inside larger tensors $\chi \times \chi$, $\chi \times \chi \times t$ if it's desirable to work with constant-size tensors for the implementation

ii) Choose environment bond dimension χ. Then repeat the steps 1 - 4 until convergence

ii.1) Contract enlarged corner



$= C2x2_i \in \mathrm{Matrix}_{(\chi t)\times(\chi t)}$

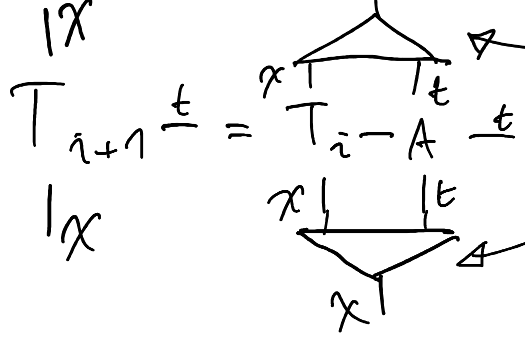
ii.2) Diagonalize enlarged corner by eigenvalue decomposition

$$C2x2_i = U_i \Lambda_i U_i^{\dagger}$$

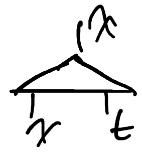
ii.3) Compress enlarged corner by keeping leading χ eigenvalues (by magnitude i.e. $|\Lambda_j| \geq |\Lambda_k|$ for $j<k$). Compress enlarged half-row/-column tensor by projectors formed from leading χ eigenvectors in U.

$$C_{i+1} = \Lambda_i[:\chi]$$

(or diagonal matrix $\chi \times \chi$ with $\Lambda_{[:\chi]}$ on diagonal)



Projector with leading χ columns $P = U_{\chi t \times}[:\chi]$ reshaped as a rank-3 tensor with dimensions $\chi \times t \times \chi$



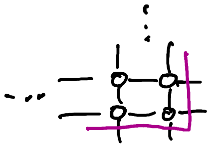
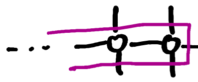
ii.4) Normalize tensors, for example by largest element (in magnitude)

$$C_{i+1} = C_{i+1} / \max(|C_{i+1}|) \ \& \ T_{i+1} = T_{i+1} / \max(|T_{i+1}|)$$

[here $|.|$ is element-wise absolute value]

check convergence, by i.e. measuring the size of the difference of $\Lambda_{i+1} - \Lambda_i$.

iii) Converged tensors C, T approximate infinite

quadrant  $\approx C$ & half-row/-column  $\approx T$

End TL;DR summary

Let's look at the related problem:

Partition function of 2D classical stat-mech model:

→ 2D Ferromagnetic Ising model on a square lattice

$$\mathcal{H} = -\sum_{ij} \sigma_i \sigma_j \quad \to Z(\beta) = \mathrm{Tr}\, e^{-\beta \mathcal{H}}$$

with $\sigma \in \{-1, 1\}$ (Just numbers)

$$Z = \mathrm{Tr}_{\{\sigma\}} \exp\Big(+\beta \sum_{\langle ij \rangle} \sigma_i \sigma_j\Big)$$

$$= \mathrm{Tr}_{\{\vec{\sigma}\}} \prod_{\langle ij \rangle} \exp(\beta \sigma_i \sigma_j)$$

$$= \underbrace{\mathrm{Tr}_{\{\vec{\sigma}\}}}\ \underbrace{\prod_{\langle ij\rangle} \exp(\beta\sigma_i\sigma_j)}$$

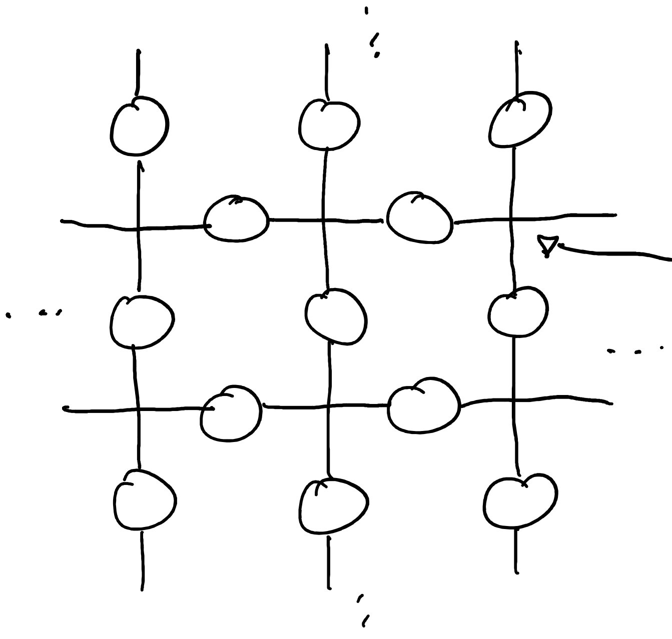
Sum over all possible configurations of spins $\vec{\sigma} \in \{+1,-1\}^{\otimes L^2}$ on a square lattice of size $L \times L$

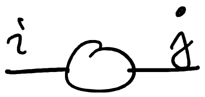
Given a configuration $\vec{\sigma} = (\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_{L^2})$ it's Boltzmann weight $\exp(-\beta H(\sigma))$ is expressed as a product of weights assigned to the edges of square lattice while spins $\sigma_1, \sigma_2, \dots, \sigma_{L^2}$ sit on vertices

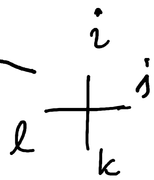
$\downarrow$ graphical interpretation

Construction of Z as tensor network contraction (I.3.1)

$Z =$




$= \exp(\beta\sigma_i\sigma_j)$ 2x2 matrix

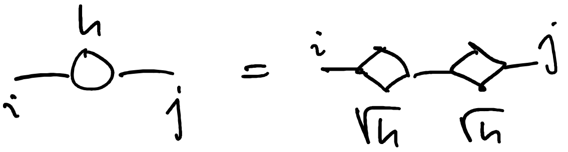

$= \delta_{ijkl}$ rank-4 $2\times2\times2\times2$ tensor

where only non-zero elements of delta are $\delta_{0000} = \delta_{1111} = 1$

For any particular choice of spins states $\Leftrightarrow$ configuration we get its Boltzmann weight from construction.

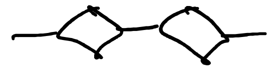
Let's first turn this network into familiar PEPS format where tensors sit in the vertices of the lattice, not on edges.

i) express Boltzman weights as a product of their square root

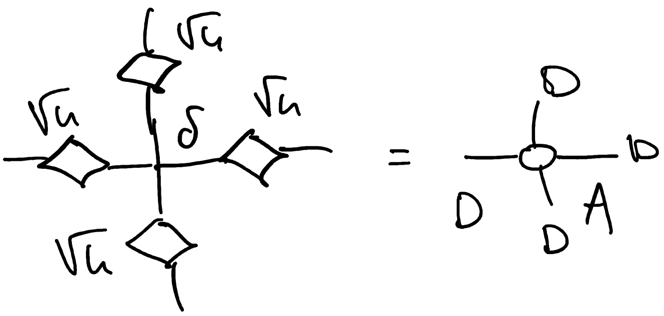


$h_{ij} = \exp(\beta \sigma_i \sigma_j)$

$$= \begin{array}{cc|cc} & \sigma_i \backslash \sigma_j & \text{index} & 0 & 1 \\ & & \text{state} & +1 & -1 \\ \text{index} & \text{state} & & & \\ 0 & +1 & & e^{\beta} & e^{-\beta} \\ 1 & -1 & & e^{-\beta} & e^{\beta} \end{array}$$

ii) Replacing  on every edge by 

we can absorb $\sqrt{h}$'s symmetrically into δ's on each vertex:

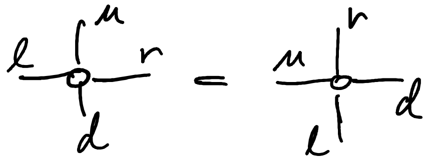


where $D=2$

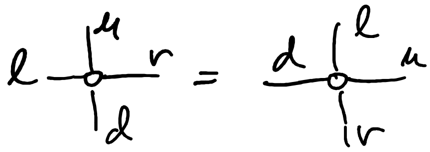
or in Einstein summation convention

$$A_{uldr} = \sum_{u'l'd'r'} (\sqrt{h})_{uu'} (\sqrt{h})_{ll'} (\sqrt{h})_{dd'} (\sqrt{h})_{rr'} \, \delta_{u'l'd'r'}$$

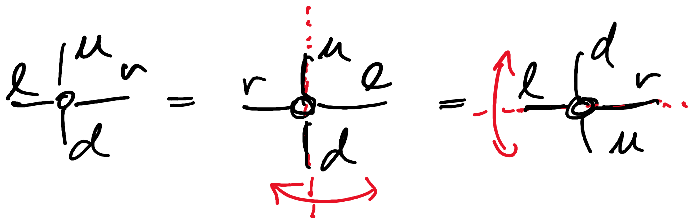
Note: tensor is symmetric under the permutations of it's indices corresponding to point-group symmetries of square lattice

i) 

counter-clockwise rotation by $\pi/2$

ii) 

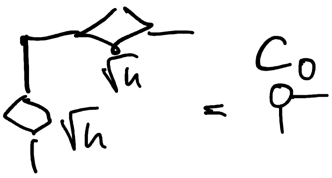
clockwise rotation by $\pi/2$

iii) 

reflections along y-axis and x-axis.

Therefore, we have expressed the partition function as a contraction of a PEPS TN generated by tensor $\underline{a}$.

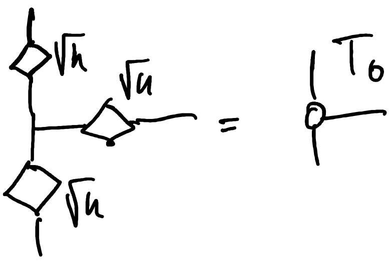
For open boundary condition, expressed through tensors:

*) 

where $= \delta_{ij}$ with non-zero entries $\delta_{00}=\delta_{11}=1$

C_0 is $D\times D$ tensor.

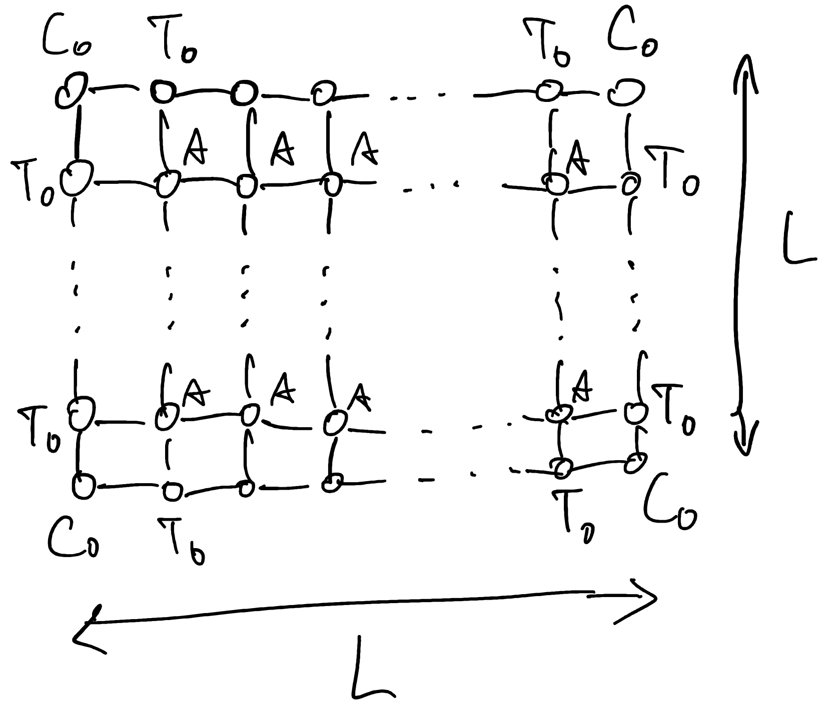
&

*) 

where $= \delta_{ijk}$ with entries $\delta_{000}=\delta_{111}=1$

T_0 is $D\times D\times D$ tensor

The partition function of $L \times L$ system is given by the contraction of

$Z_L =$ 

All contracted indices (lines) have dimension $D=2$.

Exact contraction of such network is exponentially costly, scaling as $\propto D^{L+2}$.

Therefore, to move forward we have to abandon exact contraction and instead try to approximate the result

The central idea of the approach outlined below is the real-space renormalization group:

Corner transfer matrix renormalization group

Baxter 50's, Nishino & Okunishi '96

Let's constructively perform this variant of RG procedure.

Start with a partition function for a small system, say $L=3$:

$$Z_3 = \begin{array}{ccccc} C_0 & - & T_0 & - & C_0 \\ | & & | & & | \\ T_0 & - & a & - & T_0 \\ | & & | & & | \\ C_0 & - & T_0 & - & C_0 \end{array} =$$

This is a very small network & we can easily contract it giving the result:

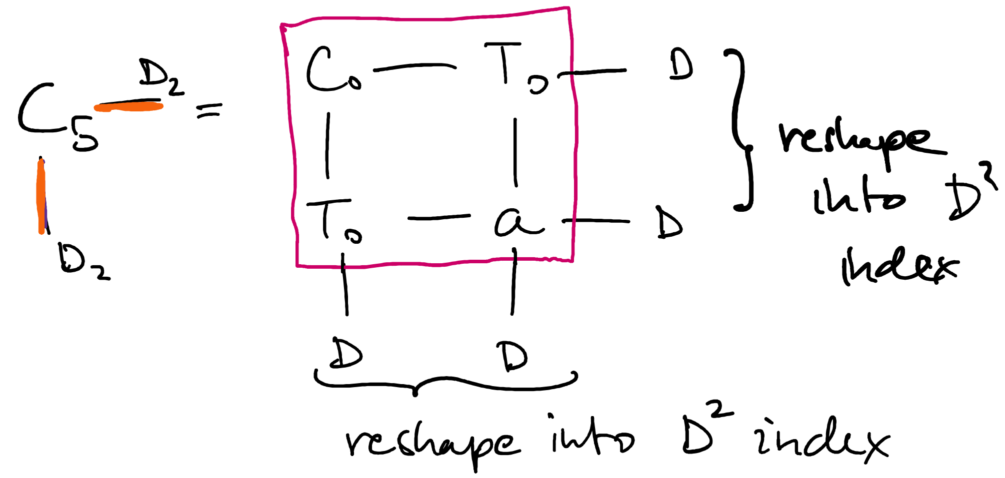
Let's consider a larger network

$$Z_5 = \begin{array}{ccccccccc} C_0 & - & T_0 & - & T_0 & - & T_0 & - & C_0 \\ | & & | & & | & & | & & | \\ T_0 & - & a & - & a & - & a & - & T_0 \\ | & & | & & | & & | & & | \\ T_0 & - & a & - & a & - & a & - & T_0 \\ | & & | & & | & & | & & | \\ T_0 & - & a & - & a & - & a & - & T_0 \\ | & & | & & | & & | & & | \\ C_0 & - & T_0 & - & T_0 & - & T_0 & - & C_0 \end{array}$$

Again, this network can be contracted exactly since it's still not too large

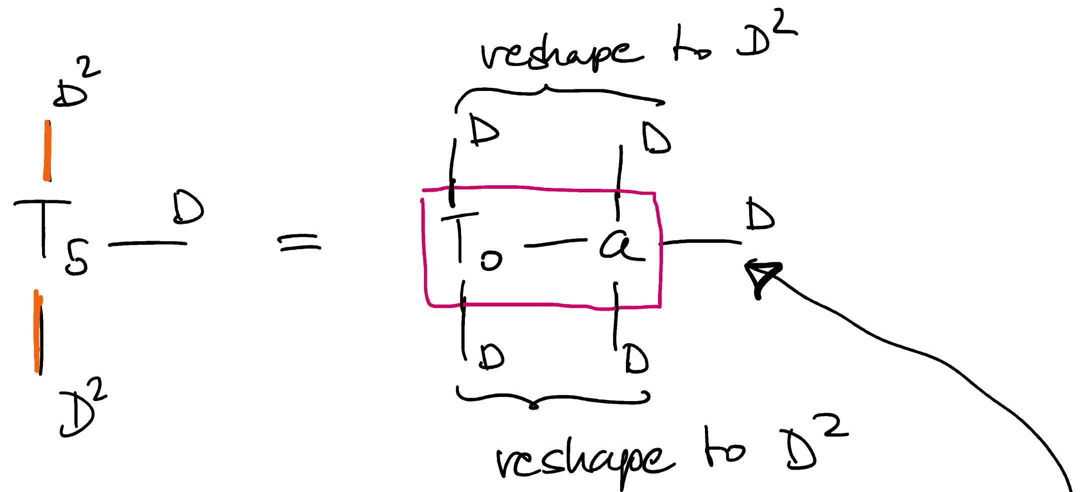
Let us rewrite this TN using a new set of tensors

i) Corner



which is obtained by contraction of enlarged corner of the Z_5 network. It's a rank-2 tensor with dimensions $D^2 \times D^2$.

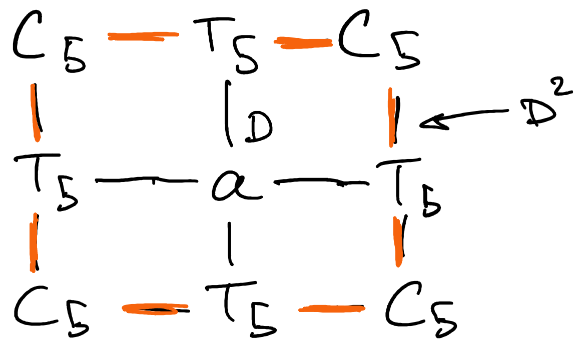
ii) half-row/-column tensor



which is a rank-3 tensor with dimensions $D^2 \times D^2 \times D$ (let's take this index to be the last index of T_5)

In terms of C_5 & T_5, the TN Z_5 can be rewritten as:

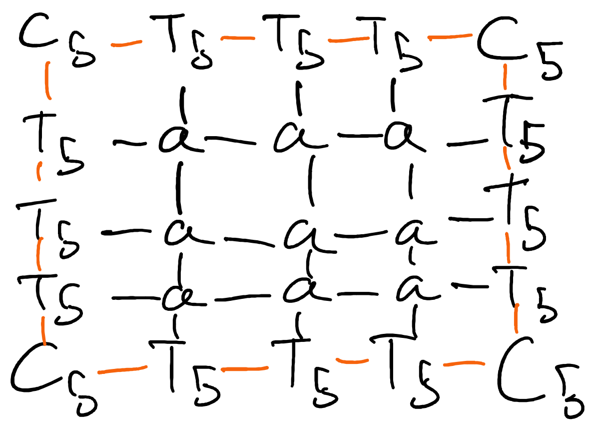
$Z_5 =$



This rewriting is exact. We merely packaged/accumulated some of the contractions into a larger tensors C_5, T_5 with their indices on the edge (orange) having dimension D^2.

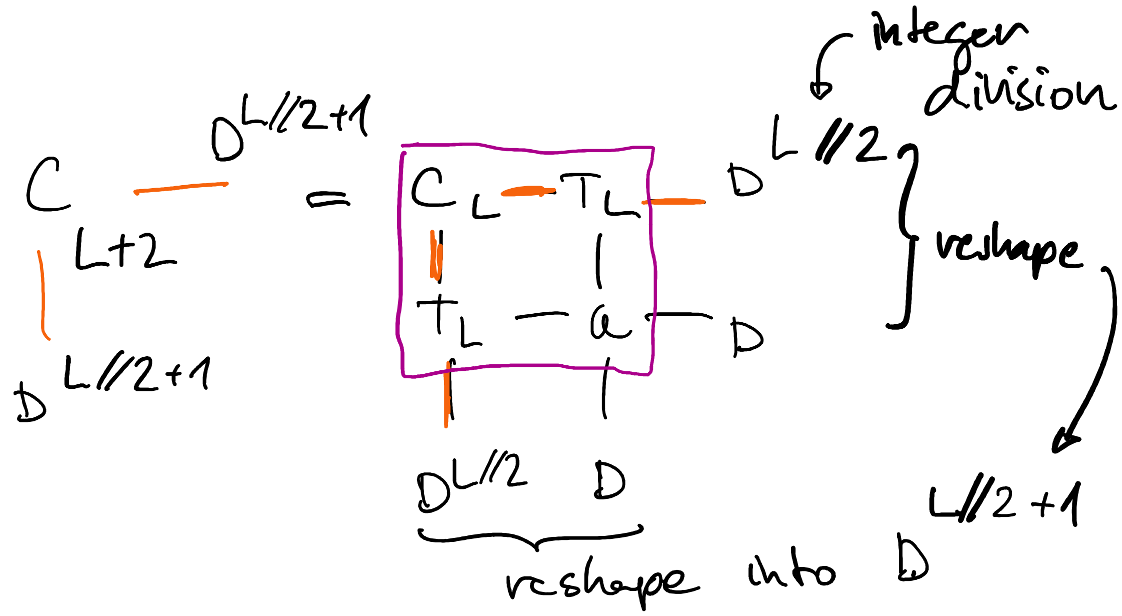
Let's enlarge the system further, to $L=9$ and write the TN using C_5, T_5:

$Z_9 =$

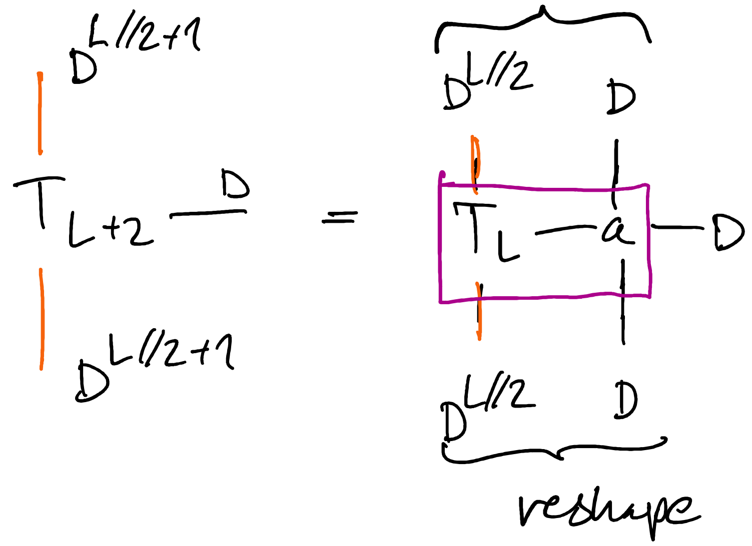


Once again let's construct a new corner and half-row/column tensor, this time starting from an arbitrary C_L, T_L

i)

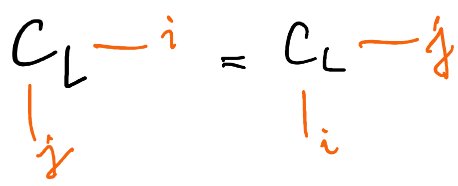


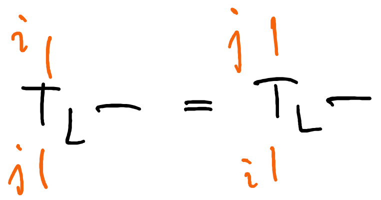
ii)



For current system $L=9$, C_9 is $D^3 \times D^3$ matrix and T_9 is $D^3 \times D^3 \times D$ rank-3 tensor.

Note: The symmetries of tensor $\underline{a}$ carry over to tensors C_L, T_L. In particular





$\Rightarrow$ C_L is a symmetric tensor

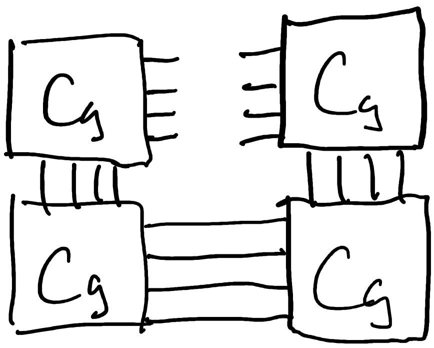
$\Rightarrow$ T_L is symmetric wrt. to exchange of edge indices.

Therefore, once again Z_g can be expressed exactly as a contraction of TN with even larger tensors C_g, T_g

$$Z_g = \begin{array}{ccccc} C_g & - & T_g & - & C_g \\ | & & |^{D} & & | \leftarrow D^3 \\ T_g & - & a & - & T_g \\ | & & | & & | \\ C_g & - & T_g & - & C_g \end{array}$$

Continuing this process, we will eventually again hit the exponential wall, since the size of the C_L, T_L grows as $\propto D^{L/2}$.

Let's make an observation:

$$\begin{array}{ccc} C_g & - \; - & C_g \\ | & & | \\ C_g & — & C_g \end{array} = \rho_g =$$ 

ρ_g is a density matrix for 4 spins living on the vertical half-cut of the system.

Then $Z_g = \mathrm{Tr}\left(C_g^4\right)$

or more generally

$$\begin{array}{ccc} C_L & — & C_L \\ | & & | \\ C_L & — & C_L \end{array} = \rho_{L/2}$$

is density matrix for $L/2$ spins on vertical half-cut.

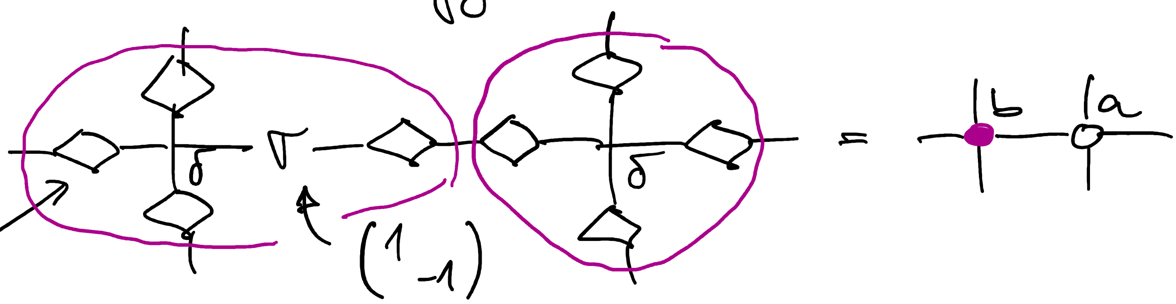
and $Z_{L-1} = \mathrm{Tr}\left(C_L^4\right)$

Note on the observables

To measure expectation value of spin σ sitting on such cut

$$\langle \sigma \rangle = \frac{\mathrm{Tr}\ \sigma \exp(-\beta H)}{\mathrm{Tr}\ \exp(-\beta H)}$$

we should modify the on-site tensor:

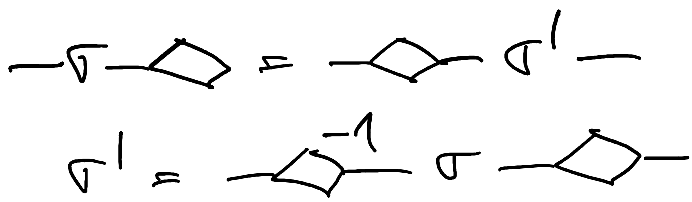


$\begin{pmatrix} 1 & \\ & -1 \end{pmatrix}$

$= \sqrt{h}$

$= \sqrt{\begin{pmatrix} e^{\beta} & e^{-\beta} \\ e^{-\beta} & e^{\beta} \end{pmatrix}}$

Alternatively, we can instead transform the observables:



To evaluate physical observables on vertical half-cut, transform them by and then evaluate transformed observables on $S_{L/2}$.

Given that C_L is symmetric, we can simply diagonalize it: $C_L = U_L \Lambda_L U_L^+$

$$* \boxed{\rho_{L/2} = U_L \Lambda_L^4 U_L^+}$$

and

$$* \; Z_{L-1} = \mathrm{Tr}\left([U_L \Lambda_L U_L^+]^4 \right) = \mathrm{Tr}(\Lambda^4)$$

since C_L is $D^{L/2} \times D^{L/2}$ matrix,

$$\boxed{Z_{L-1} = \sum_{i=1}^{D^{L/2}} \Lambda_i^4}$$

Now we can turn back to the problem of C_L, T_L growing exponentially large with L.

(I.3.3) Core proposition – variational compression:

To avoid exponential growth, let's compress the tensors C_L, T_L to size at most

χ – the environment (edge) dimension

$$\begin{matrix} C_L & \xrightarrow{\text{compress}} & \tilde{C}_L^{(\chi)} & \chi \times \chi \text{ matrix} \\ T_L & & \tilde{T}_L^{(\chi)} & \chi \times \chi \times D \text{ tensor} \end{matrix}$$

We want this compression to be optimal, such that the error on the corresponding density matrix $\rho_{L/2}$ and hence all observables it encodes is minimal. We define the error wrt. Frobenius norm

due to $\rho_{L/2} = C_L^4$

$$\boxed{\min_{\tilde{\rho}} \left| \rho_{L/2} - \tilde{\rho}^{(\chi)}_{L/2} \right|_F^2 \Longleftrightarrow \min_{\tilde{C}} \left| C_L - \tilde{C}_L^{(\chi)} \right|_F^2}$$

Optimal solution of this problem (low-rank matrix approximation) is given by truncated SVD, or in this case symmetric eigendecomposition

$$\rho_{L/2} = C_L^4 = U_L \Lambda_L^4 U_L^\dagger \quad \text{where } \forall i,j \;\; j>i \;\; |\Lambda_i| \geq |\Lambda_j|$$

$$\Rightarrow \tilde{\rho}^{(\chi)}_{L/2} = P \Lambda^4_{[:\chi]} P^\dagger \quad \text{where } P = U_{[:,:\chi]}$$

leading χ columns

Note: this compression preserves the symmetries of $\tilde{C}, \tilde{T}$

This gives an error

$$\text{error}_C(\chi) = \left| C_L - \tilde{C}_L^{(\chi)} \right|_F^2 = \sum_{i=1}^{D^{L/2}} \Lambda_i^2 - \sum_{i=1}^{\chi} \Lambda_i^2 = \sum_{i=\chi+1}^{D^{L/2}} \Lambda_i^2$$

Hence for the partition function:

$$\text{error}_Z(\chi) = Z_{L-1} - \tilde{Z}_{L-1} = \mathrm{Tr}\left(C_L^4 - [\tilde{C}_L^{(\chi)}]^4\right) = \sum_{i=\chi+1}^{D^{L/2}} \Lambda_i^4$$

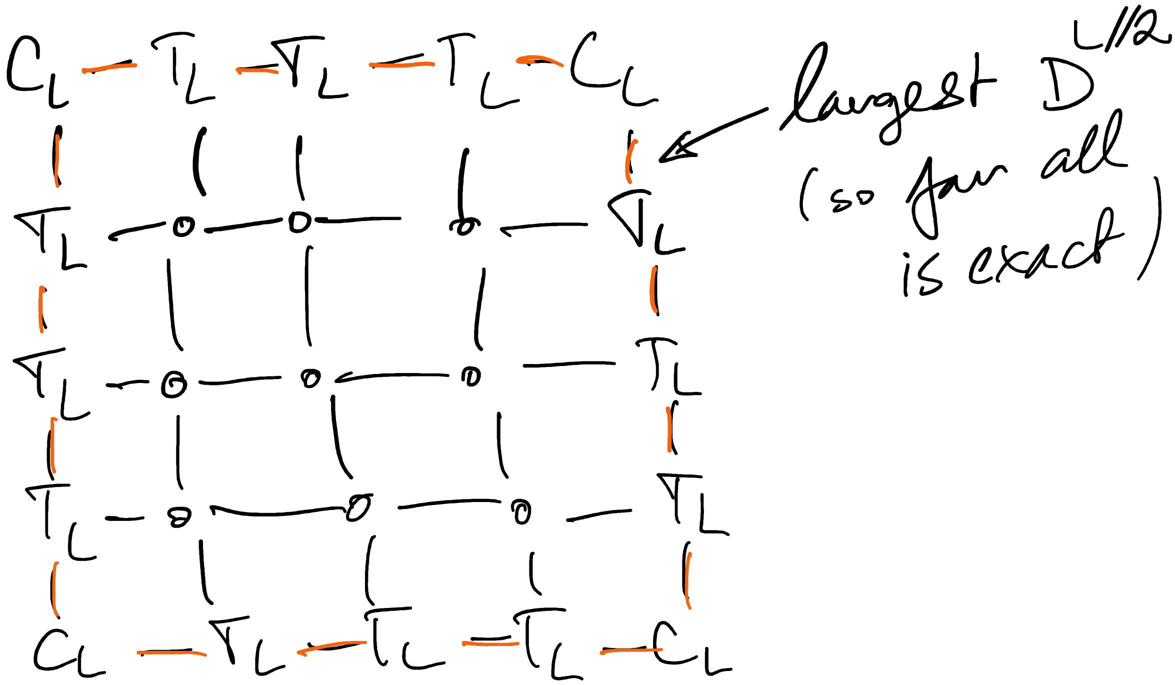
This choice is <u>systematically improvable</u> by increasing χ. It also leads to <u>variational</u> approximation of partition function.

<u>Why variational?</u> Because $\forall \chi, \chi'$ where $\chi > \chi'$ the error$(\chi') \leq$ error(χ). Given that thermodynamic free energy is $F = -\frac{1}{\beta} \log Z$, one approaches the minimal free energy from above as χ is growing.

[System in thermal equilibrium at constant volume minimizes its free energy.]

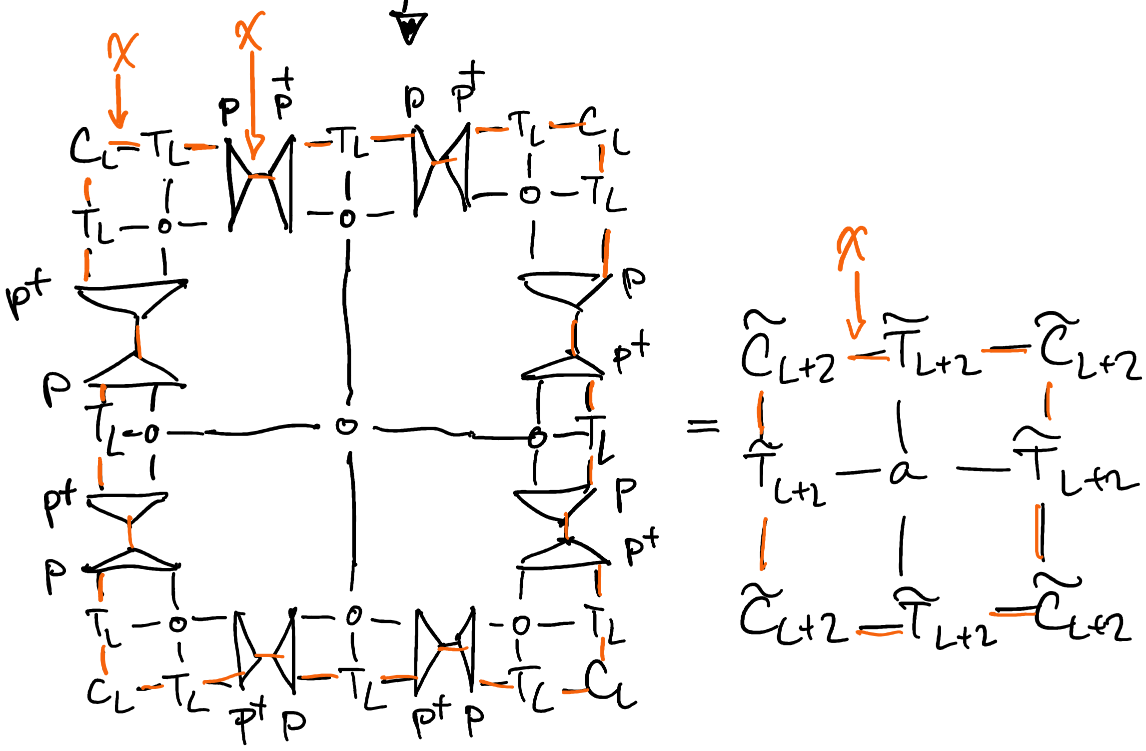
Renormalization / Compression step (I.3.4)

Once we have reached largest size L allowed by computational resources, we can compress the network as in the step above

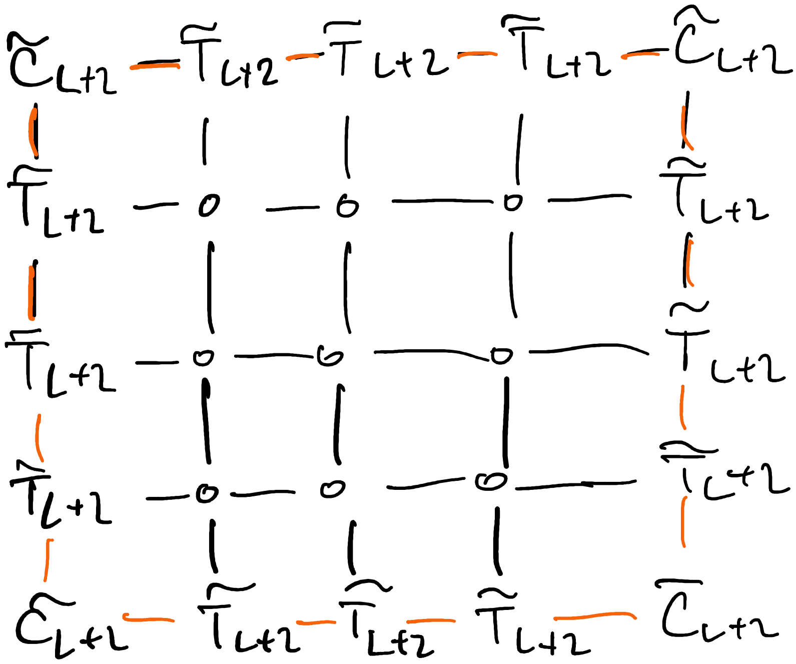


Compress

Compress



Grow



Compress

...

From this point onwards we just keep growing & compressing (RG step) never increasing environment dimension beyond χ, until we reach fixed point C^∞, T^∞.

* For fixed χ, the compression error depends on the decay of Λ's, which in turn depends on 'on-site' tensor $\underline{\underline{a}}$.

If $\chi \ll D^L$ while doing error no larger than desired ε, we avoided exponential scaling.

Finite precision & intensive quantities I.3.5

The RG steps require controlling the magnitude of $\tilde{C}, \tilde{T}$ i.e. by rescaling them. Otherwise finite precision errors would follow in numerical implementations.

Such rescaling changes the value of $Z = \mathrm{Tr}(\tilde{C}^4)$

Still, intensive quantities such as local observables or <u>partition function per site</u> are unambiguous:

$$Z_L = \begin{array}{ccc} C_L & - & C_L \\ | & & | \\ C_L & - & C_L \end{array} =: \mathcal{Z}^{L^2}$$

← Definition of partition function per site

$$Z_{L+1} = \begin{array}{ccccc} C_L & - & T_L & - & C_L \\ | & & | & & | \\ T_L & - & \circ & - & T_L \\ | & & | & & | \\ C_L & - & T_L & - & C_L \end{array} =: \mathcal{Z}^{(L+1)^2}$$

$$Z_{L\times(L+1)} = \begin{array}{ccc} C_L & - & C_L \\ | & & | \\ T_L & - & T_L \\ | & & | \\ C_L & - & C_L \end{array} =: \mathcal{Z}^{L\times(L+1)}$$

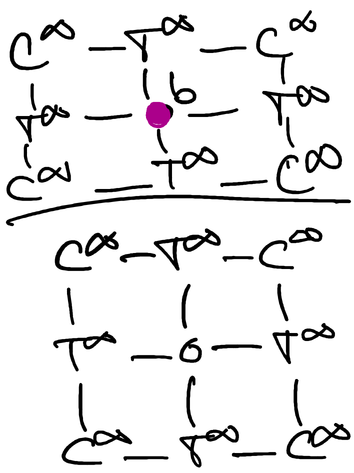
(assymetric system)

$$\boxed{\mathcal{Z} = \frac{Z_{L+1} \cdot Z_L}{Z_{L\times(L+1)} \cdot \underbrace{Z_{(L+1)\times L}}_{\text{different in case on no } C_{4v} \text{ symmetry}}}}$$

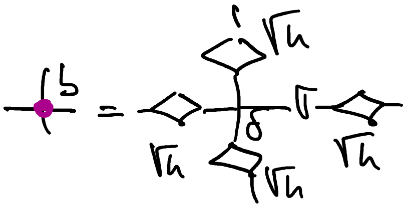
This formula remains valid even for compressed fixed point C^∞, T^∞, where strictly speaking size of the system is no longer well-defined.

<u>On-site magnetization</u> can be estimated following its definition:

$$\langle\sigma\rangle_x = \frac{\mathrm{Tr}(\sigma \exp(-\beta H))}{\mathrm{Tr}\exp(-\beta H)} =$$

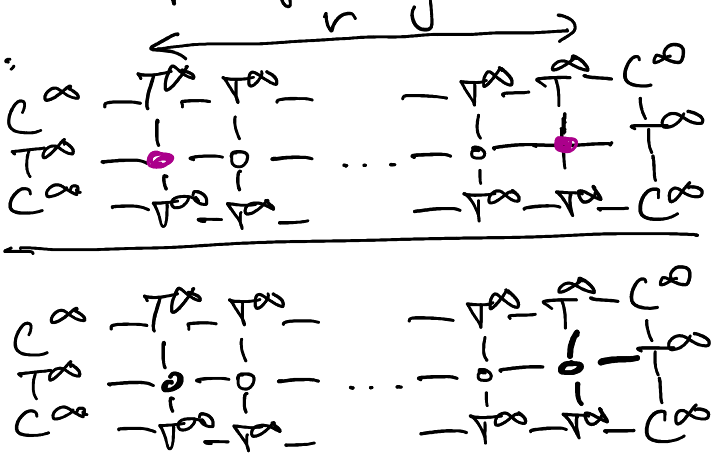


where

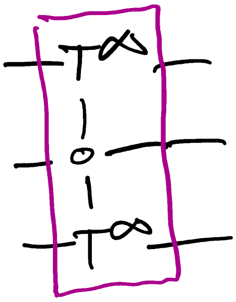


<u>Correlation length</u> is derived from 2-point correlation function or specifically its TN approximation:

$$\langle \sigma_0 \sigma_r \rangle =$$

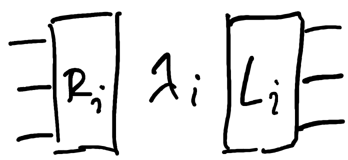


which reduces to analogous argument as for 1D MPS, here



playing the role of $\chi^2 D \times \chi D^2$ $\mathbb{T}$-transfer matrix.

It's gap, i.e., for $\mathbb{T} = \sum_{i=1}^{\chi^2 D}$



determines correlation length as $\xi_i = -\frac{1}{\log|\lambda_i/\lambda_0|}$.