Tensor Network Approach to Real-Time Path Integral

Shinji Takeda

(Kanazawa U.)

Frontiers in Lattice QCD and related topics

2019.04.15-26 @YITP

Contents

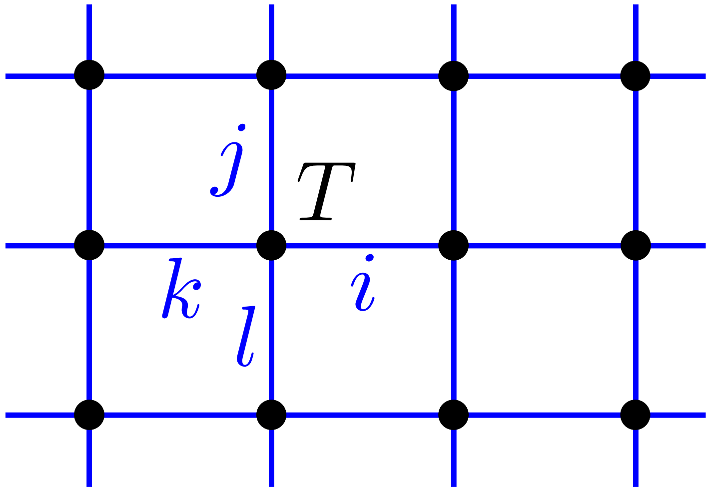
- Introduction to tensor network approach
  - Why/What's tensor network
  - Lagrangian/path integral approach
  - Tensor renormalization group (TRG)

- Real-time path integral by Tensor network
  - example: 1+1 lattice scalar field theory
  - Rewrite path integral by Tensor network representation
  - numerical results (free case)

Why tensor networks?

- Success of Monte Carlo (MC) methods in various fields
- But, MC suffers from Sign problem
  - e.g. QCD+μ, θ-term, chiral gauge theory, lattice SUSY, real-time path integral,…)
- Tensor network is free from Sign problem
- Because Probability is not used!

What's tensor network?

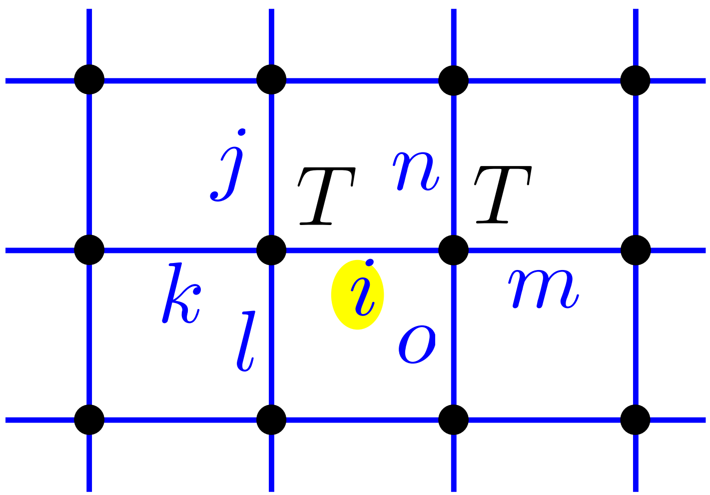


$$T_{ijkl}$$

tensor : lattice point

indices : link

What’s tensor network?



$$\Leftrightarrow \sum_{\cdots,i,\cdots} \cdots T_{ijkl} T_{mnio} \cdots$$

contraction

A target quantity (wave function/partition function) is represented by tensor network

Tensor network approaches

Hamiltonian/Hilbert space	**Lagrangian/Path integral**
Quantum many-body system	Classical many-body system/path integral rep. of quantum system
Wave function of ground state/excited states	Partition function/correlation functions
Variational method	Approximation, Coarse graining
Real time, Out-of-equilibrium, Quantum simulation	Useful in equilibrium system suffering from the sign problem in MC(QCD+μ, etc.)
DMRG, MPS, PEPS, MERA, …	TRG, SRG, HOTRG, TNR, Loop-TNR, …

Tensor network approaches

Hamiltonian/Hilbert space	Lagrangian/Path integral
Quantum many-body system	Classical many-body system/path integral rep. of quantum system
Wave function of ground state/excited states	Partition function/correlation functions
Variational method	Approximation, Coarse graining
Real time, Out-of-equilibrium, Quantum simulation	Useful in equilibrium system suffering from the sign problem in MC(QCD+μ, etc.)
DMRG, MPS, PEPS, MERA, …	TRG, SRG, HOTRG, TNR, Loop-TNR, …

↘ My talk

Tensor network rep. of Z

Levin & Nave 2007

$$Z = \int [d\phi] \exp\left[-S(\{\phi\})\right]$$

↓

Target

Tensor network rep. of Z

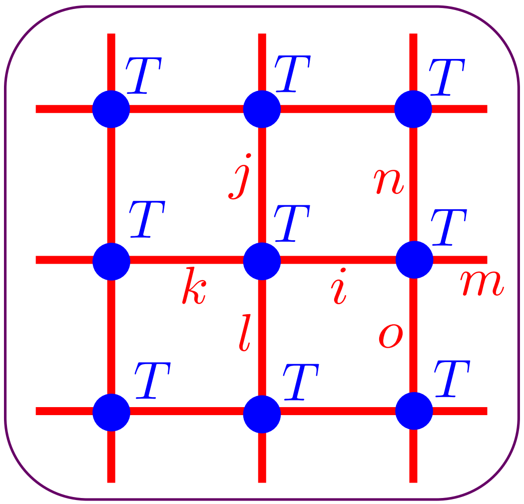
Levin & Nave 2007

Rewrite the partition function in terms of contractions of tensors

$$Z = \int [d\phi] \exp\left[-S(\{\phi\})\right]$$

$$= \sum_{\dots,i,j,k,l,\dots} \cdots T_{ijkl} T_{mnio} \cdots$$

tensor network representation in 2D system



in 2D system

tensor : lives on a lattice site

index : lives on a link

uniform : all tensors are the same

elements of tensor : model-dependent

Tensor network rep. of Z

e.g. 2D Ising model

$$Z = \sum_{\{s\}} \exp\left(\sum_{<x,y>} \beta s_x s_y \right) = \sum_{\ldots,i,j,k,l,\ldots} \cdots T_{ijkl} T_{mnio} \cdots$$

Tensor network rep. of Z

e.g. 2D Ising model

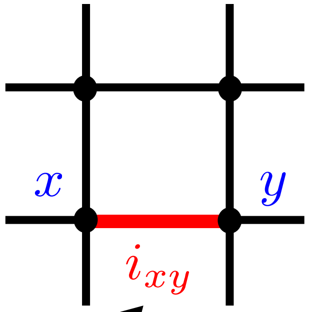
$$Z=\sum_{\{s\}}\exp\left(\sum_{<x,y>}\beta s_x s_y\right)=\sum_{\ldots,i,j,k,l,\ldots}\cdots T_{ijkl}T_{mnio}\cdots$$

1) Expand Boltzmann weight as in High-T expansion

2) Identify integer, which appears in the expansion, as new d.o.f. → index of tensor

$$\begin{aligned} e^{\beta s_x s_y} &= \cosh(\beta s_x s_y)+\sinh(\beta s_x s_y) \\ &= \cosh\beta + s_x s_y \sinh\beta \\ &= \cosh\beta(1+s_x s_y\tanh\beta) \\ &= \cosh\beta\sum_{i_{xy}=0}^{1}(s_x s_y\tanh\beta)^{i_{xy}} \end{aligned}$$

$s_x=\pm 1$



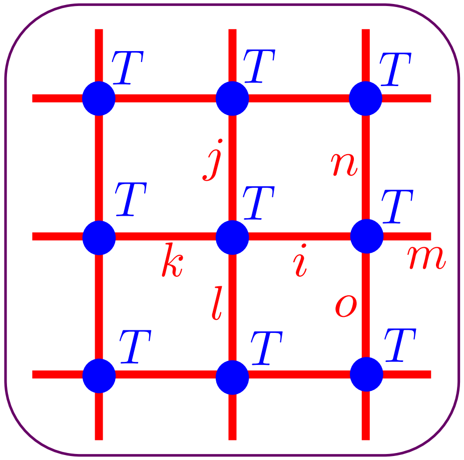
new d.o.f.

Tensor network rep. of Z

e.g. 2D Ising model

$$Z = \sum_{\{s\}} \exp\left(\sum_{<x,y>} \beta s_x s_y \right) = \sum_{\dots,i,j,k,l,\dots} \cdots T_{ijkl} T_{mnio} \cdots$$

1) Expand Boltzmann weight as in High-T expansion
2) Identify integer, which appears in the expansion, as new d.o.f. → index of tensor
3) Integrate out spin variable (old d.o.f.)
4) Get tensor network rep. !



$$\begin{bmatrix} T_{0000} & T_{0001} & T_{0010} & T_{0011} \\ T_{0100} & T_{0101} & T_{0110} & T_{0111} \\ T_{1000} & T_{1001} & T_{1010} & T_{1011} \\ T_{1100} & T_{1101} & T_{1110} & T_{1111} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \tanh\beta \\ 0 & \tanh\beta & \tanh\beta & 0 \\ 0 & \tanh\beta & \tanh\beta & 0 \\ \tanh\beta & 0 & 0 & (\tanh\beta)^2 \end{bmatrix}$$

$$\times 2(\cosh\beta)^2$$

Tensor network rep. of Z

e.g. 2D Ising model

$$Z = \sum_{\{s\}} \exp\left(\sum_{<x,y>} \beta s_x s_y \right) = \sum_{\dots,i,j,k,l,\dots} \cdots T_{ijkl} T_{mnio} \cdots$$

1) Expand Boltzmann weight as in High-T expansion
2) Identify integer, which appears in the expansion, as new d.o.f. → index of tensor
3) Integrate out spin variable (old d.o.f.)
4) Get tensor network rep. !

For every model, one has to do similar thing and the size and elements of tensor depends on the model, but the basic procedure is common for all cases

Tensor network rep. of Z

depends on property of field and interaction

- Scalar field (non-compact)
  - Orthonormal basis expansion
    Shimizu mod.phys.lett. A27,1250035(2012), Lay & Rundnick PRL88,057203(2002)
  - Gauss Hermite quadrature Sakai et al., JHEP03(2018)141
- Gauge field (compact : SU(N) etc) Meurice et al., PRD88,056005(2013)
  - Character expansion : maintain symmetry, better convergence
- Fermion field (Dirac/Majorana)
  Shimizu & Kuramashi PRD90,014508(2014), ST & Yoshimura PTEP(2015)043B01
  - Grassmann number $\theta^2=0$ -> finite sum
  - Signature originated from Grassmann nature

$$e^{\phi\theta} = 1 + \phi\theta = \sum_{n=0}^{1} (\phi\theta)^n$$

In principle, we can treat any fields

How to carry out the contractions?

$$Z = \sum_{..,i,j,k,l,m,n,o,..} \cdots T_{ijkl} T_{mnio} \cdots$$

- So far, we have just rewritten Z
- Next step is to carry out the summation
- But, naïve approach costs ∝ 2^{2V}
- Introduce approximation to reduce the cost while keeping an accuracy by summing important part in Z

How to carry out the contractions?

$$Z = \sum_{..,i,j,k,l,m,n,o,..} \cdots T_{ijkl} T_{mnio} \cdots$$

- So far, we have just rewritten Z
- Next step is to carry out the summation
- But, naïve approach costs ∝ 2^{2V}
- Introduce approximation to reduce the cost while keeping an accuracy by summing important part in Z

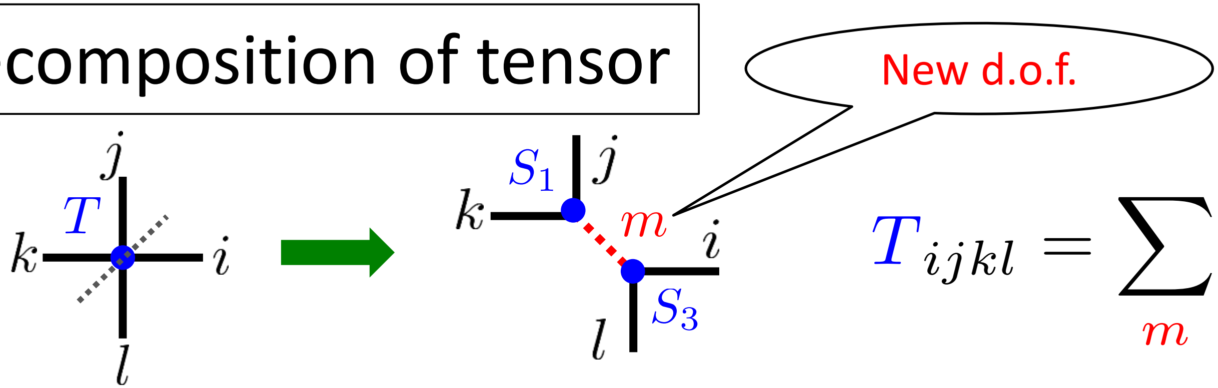
Coarse graining (renormalization, blocking)

Tensor Renormalization Group (TRG)

Levin & Nave PRL99,120601(2007)

Coarse graining (TRG)

Decomposition of tensor

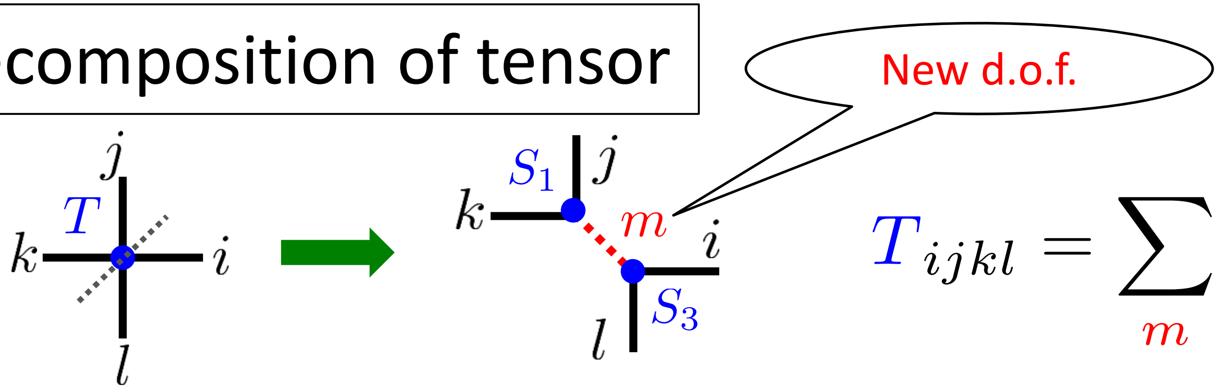


$i, j, k, l \in \{1, 2, ..., D\}$

$$T_{ijkl} = \sum_m (S_1)_{jkm} (S_3)_{lim}$$

Coarse graining (TRG)

Decomposition of tensor



$i, j, k, l \in \{1, 2, ..., D\}$

$$T_{ijkl} = \sum_m (S_1)_{jkm} (S_3)_{lim}$$

Singular value decomposition (SVD)

u, v : unitary matrix

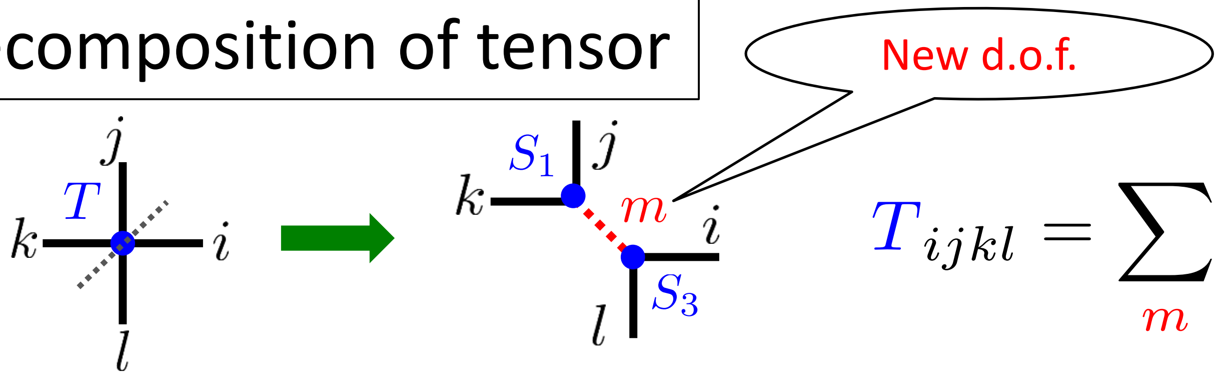
$$M_{ab} = \sum_m u_{am} \sigma_m (v^\dagger)_{mb}$$

$\sigma_1 \geq \sigma_2 \geq ... \geq 0$: singular values

$M \in \mathbb{C}^{I \times J}$

Coarse graining (TRG)

Decomposition of tensor



$i, j, k, l \in \{1, 2, ..., D\}$

$$T_{ijkl} = \sum_m (S_1)_{jkm} (S_3)_{lim}$$

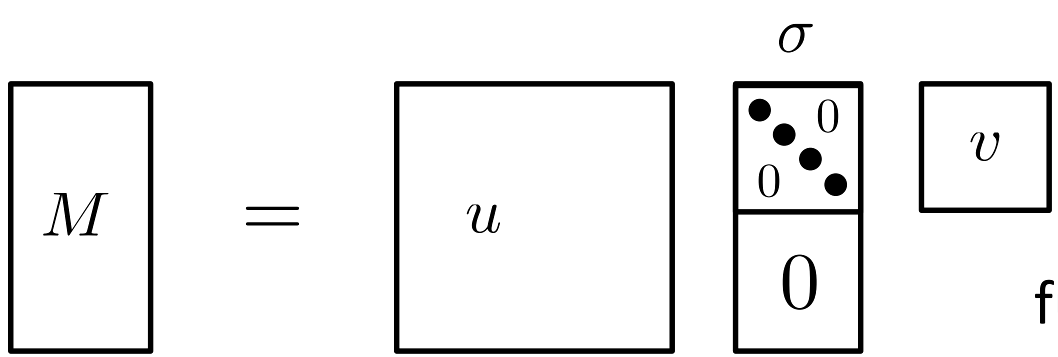
Singular value decomposition (SVD)

u, v : unitary matrix

$$M_{ab} = \sum_m u_{am} \sigma_m (v^\dagger)_{mb}$$

$\sigma_1 \geq \sigma_2 \geq ... \geq 0$: singular values

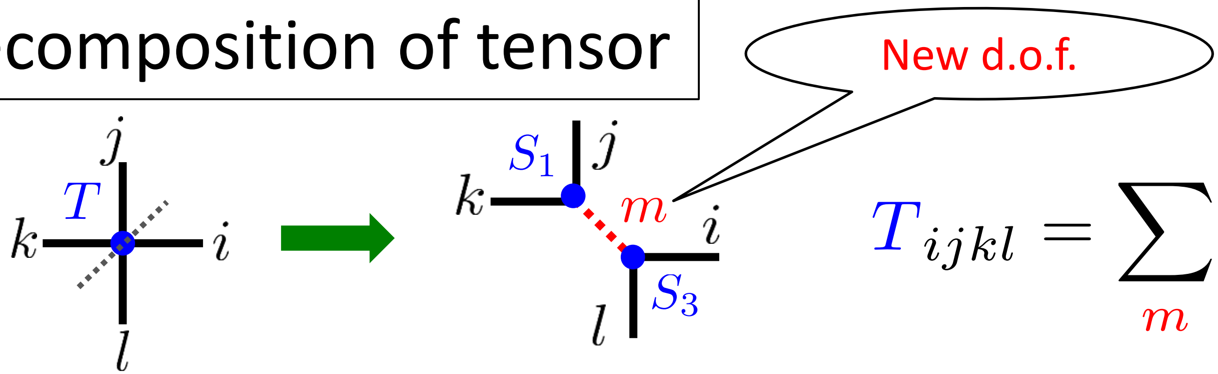
$M \in \mathbb{C}^{I \times J}$



full SVD

Coarse graining (TRG)

Decomposition of tensor



$$i, j, k, l \in \{1, 2, ..., D\}$$

$$T_{ijkl} = \sum_m (S_1)_{jkm} (S_3)_{lim}$$

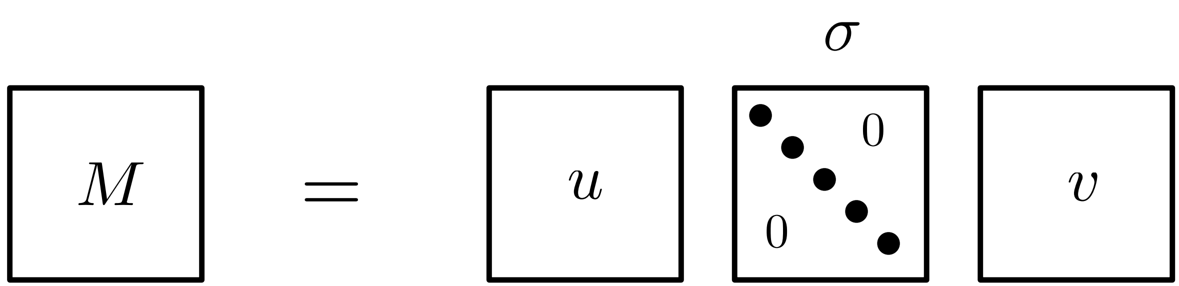
Singular value decomposition (SVD)

u, v : unitary matrix

$$M_{ab} = \sum_m u_{am} \sigma_m (v^\dagger)_{mb}$$

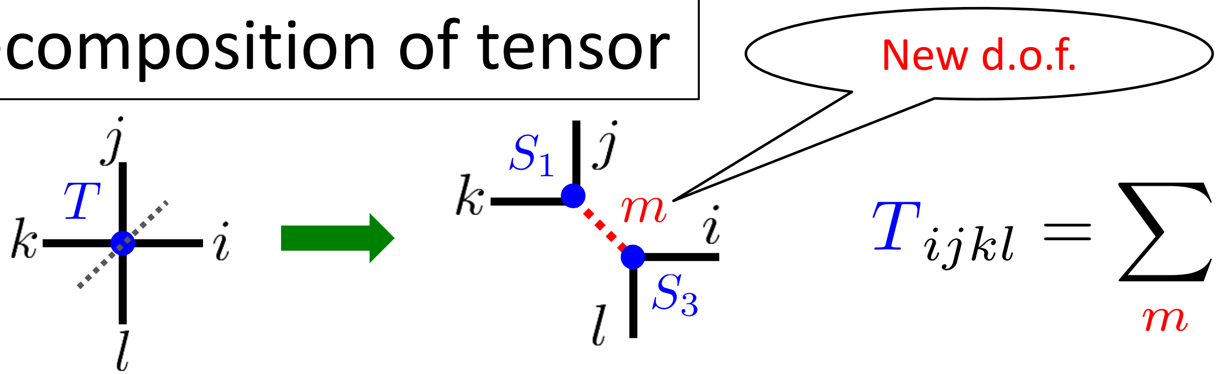
$\sigma_1 \geq \sigma_2 \geq ... \geq 0$: singular values

For square matrix



Coarse graining (TRG)

Decomposition of tensor



$$i, j, k, l \in \{1, 2, ..., D\}$$

$$T_{ijkl} = \sum_{m} (S_1)_{jkm} (S_3)_{lim}$$

Singular value decomposition (SVD)

u, v : unitary matrix

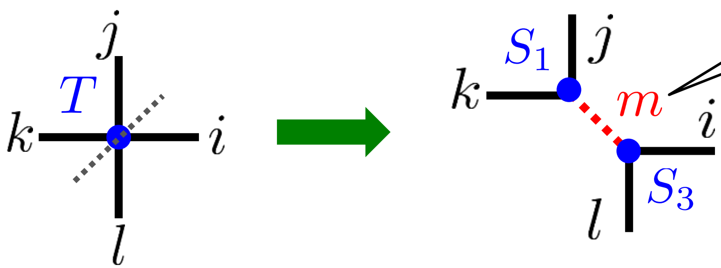
$$M_{ab} = \sum_{m} u_{am} \sigma_m (v^\dagger)_{mb}$$

$\sigma_1 \geq \sigma_2 \geq ... \geq 0$: singular values

$$T_{ijkl} = M_{(kj),(il)} = \sum_{m=1}^{D^2} u_{(kj),m} \sqrt{\sigma_m} \cdot \sqrt{\sigma_m} v^\dagger_{m,(il)}$$

Coarse graining (TRG)

Decomposition of tensor



$i, j, k, l \in \{1, 2, ..., D\}$

$$T_{ijkl} = \sum_m (S_1)_{jkm}(S_3)_{lim}$$

Singular value decomposition (SVD)

u, v : unitary matrix

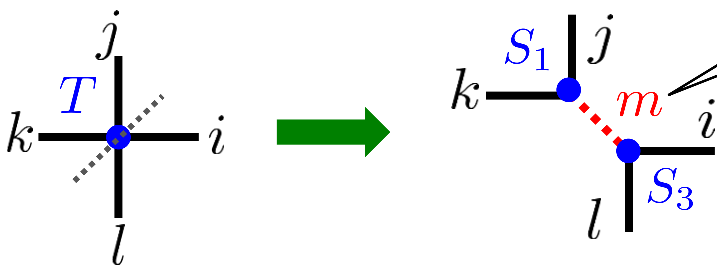
$$M_{ab} = \sum_m u_{am}\sigma_m(v^\dagger)_{mb}$$

$\sigma_1 \geq \sigma_2 \geq ... \geq 0$: singular values

$$T_{ijkl} = M_{(kj),(il)} = \sum_{m=1}^{D^2} \underline{u_{(kj),m}\sqrt{\sigma_m}} \cdot \underline{\sqrt{\sigma_m}v^\dagger_{m,(il)}} \approx \sum_{m=1}^{D_{\rm cut}} \underline{(S_1)_{jkm}}\,\underline{(S_3)_{lim}}$$

Coarse graining (TRG)

Decomposition of tensor



New d.o.f.

$i, j, k, l \in \{1, 2, ..., D\}$

$$T_{ijkl} = \sum_m (S_1)_{jkm} (S_3)_{lim}$$

Singular value decomposition (SVD)

u, v : unitary matrix

$$M_{ab} = \sum_m u_{am} \sigma_m (v^\dagger)_{mb}$$

$\sigma_1 \geq \sigma_2 \geq ... \geq 0$: singular values

$D_{\mathrm{cut}} = D$

approx. D_{cut}

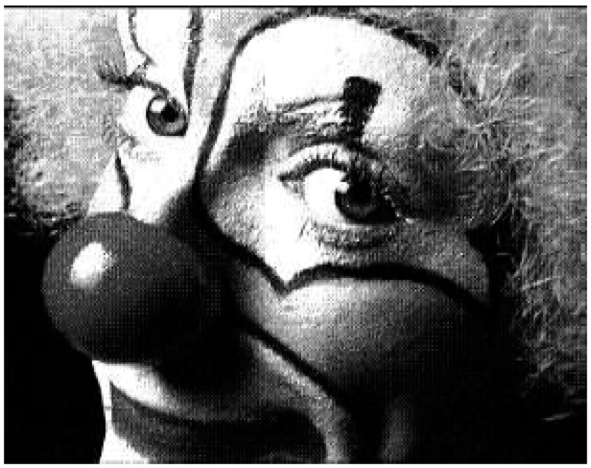
$$T_{ijkl} = M_{(kj),(il)} = \sum_{m=1}^{D^2} u_{(kj),m} \sqrt{\sigma_m} \cdot \sqrt{\sigma_m} v^\dagger_{m,(il)} \approx \sum_{m=1}^{D_{\mathrm{cut}}} (S_1)_{jkm} (S_3)_{lim}$$

Truncate at D_{cut} → Low-rank approximation → Information compression

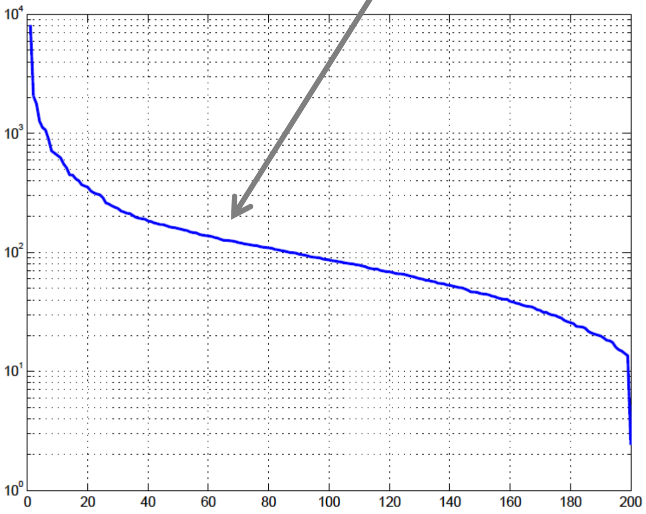
best approximation

Image compression

200 x 320 pixels B&W photograph
= 200 x 320 real matrix → $M_{ab} = \sum_m u_{am}\sigma_m (v^\dagger)_{mb}$



σ_m

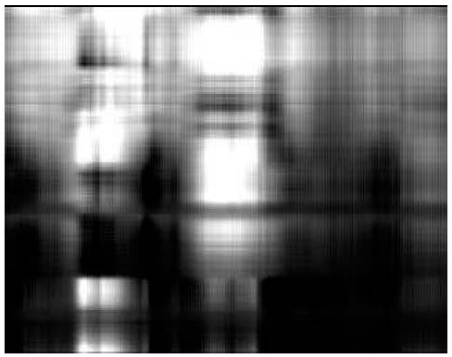


numbering m

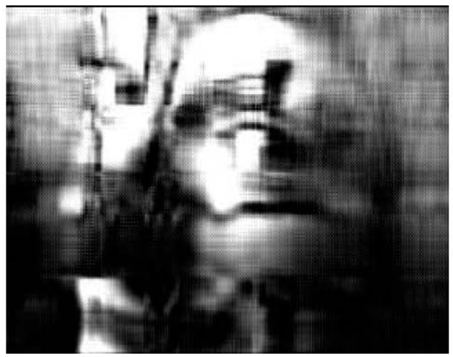
http://www.na.scitec.kobe-u.ac.jp/~yamamoto/lectures/cse-introduction2009/cse-introduction090512.PPT

Image compression

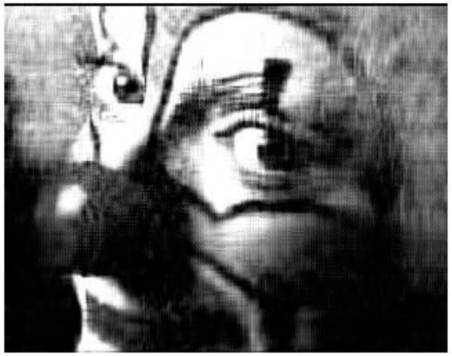
D_{cut}=3



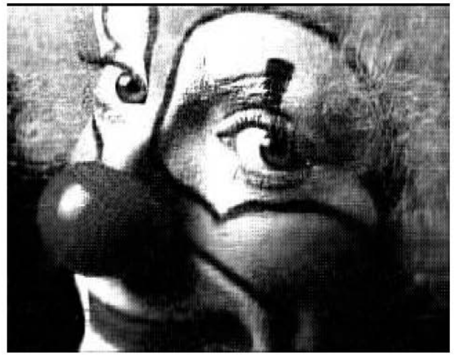
D_{cut}=10



D_{cut}=20

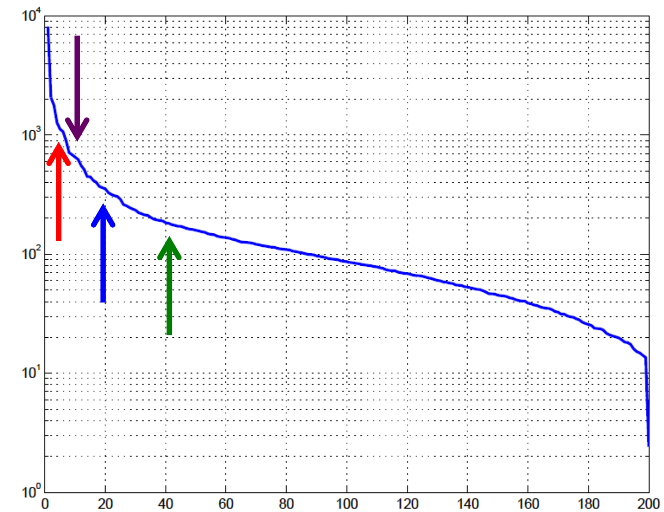


D_{cut}=40



$$M_{ab} \approx \sum_{m=1}^{D_{cut}} u_{am} \sigma_m (v^\dagger)_{mb}$$

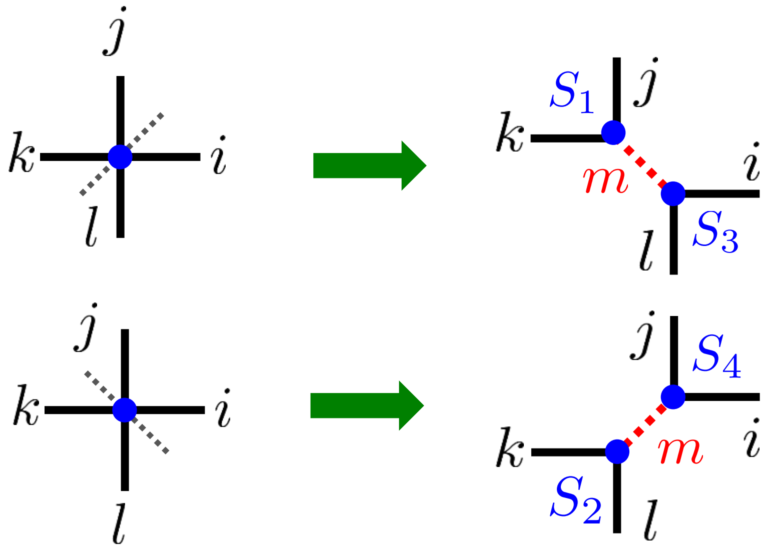
σ_m



numbering m

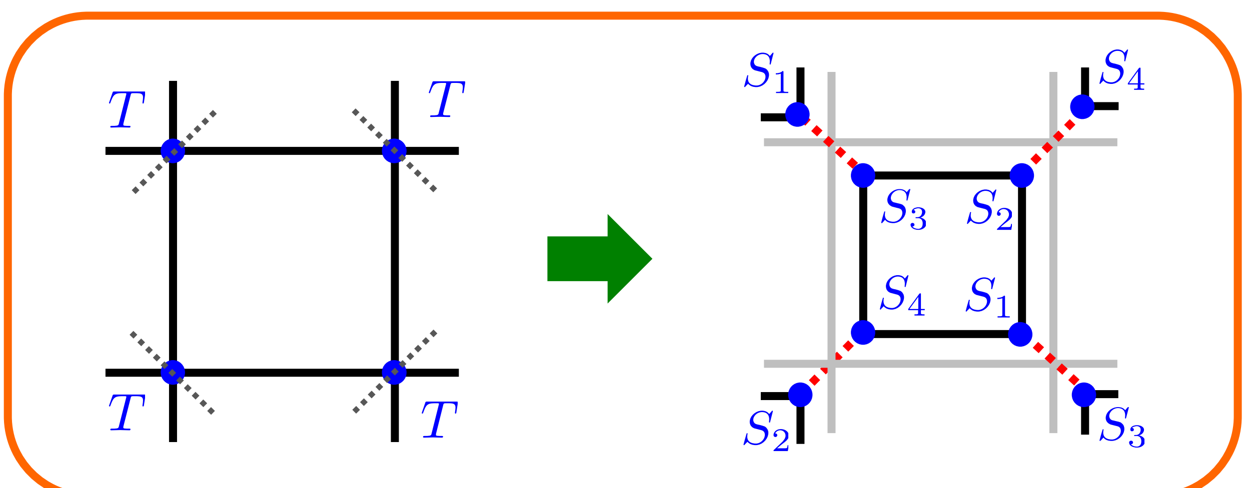
http://www.na.scitec.kobe-u.ac.jp/~yamamoto/lectures/cse-introduction2009/cse-introduction090512.PPT

Coarse graining (TRG)



$$T_{ijkl} \approx \sum_{m=1}^{D_{\mathrm{cut}}} (S_1)_{jkm} (S_3)_{lim}$$

$$T_{ijkl} \approx \sum_{m=1}^{D_{\mathrm{cut}}} (S_2)_{klm} (S_4)_{ijm}$$



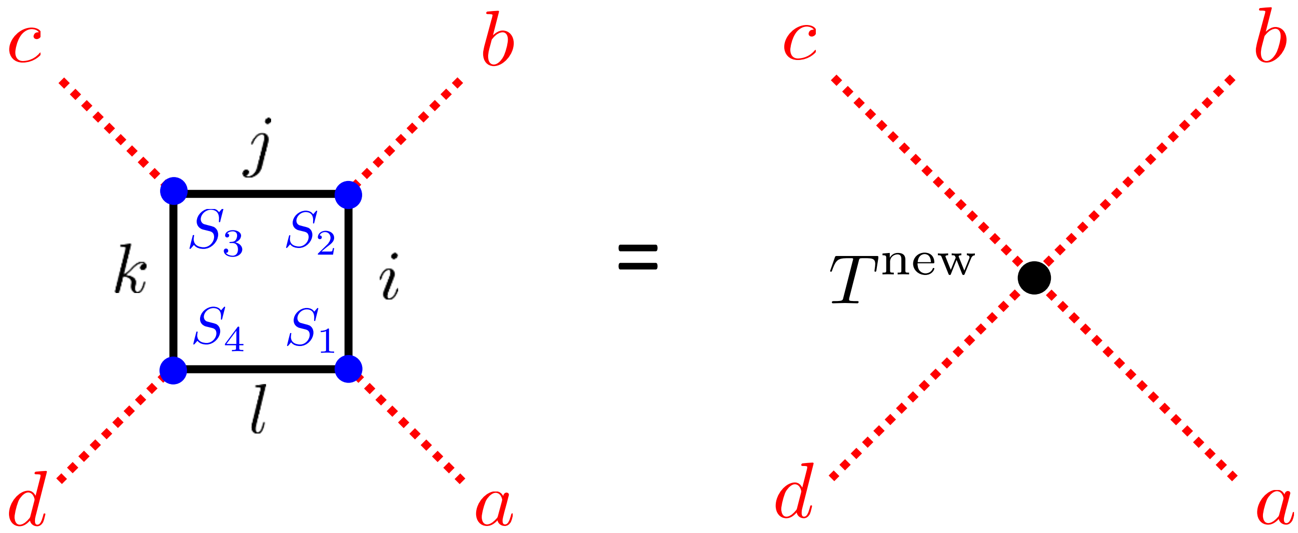
Coarse graining (TRG)

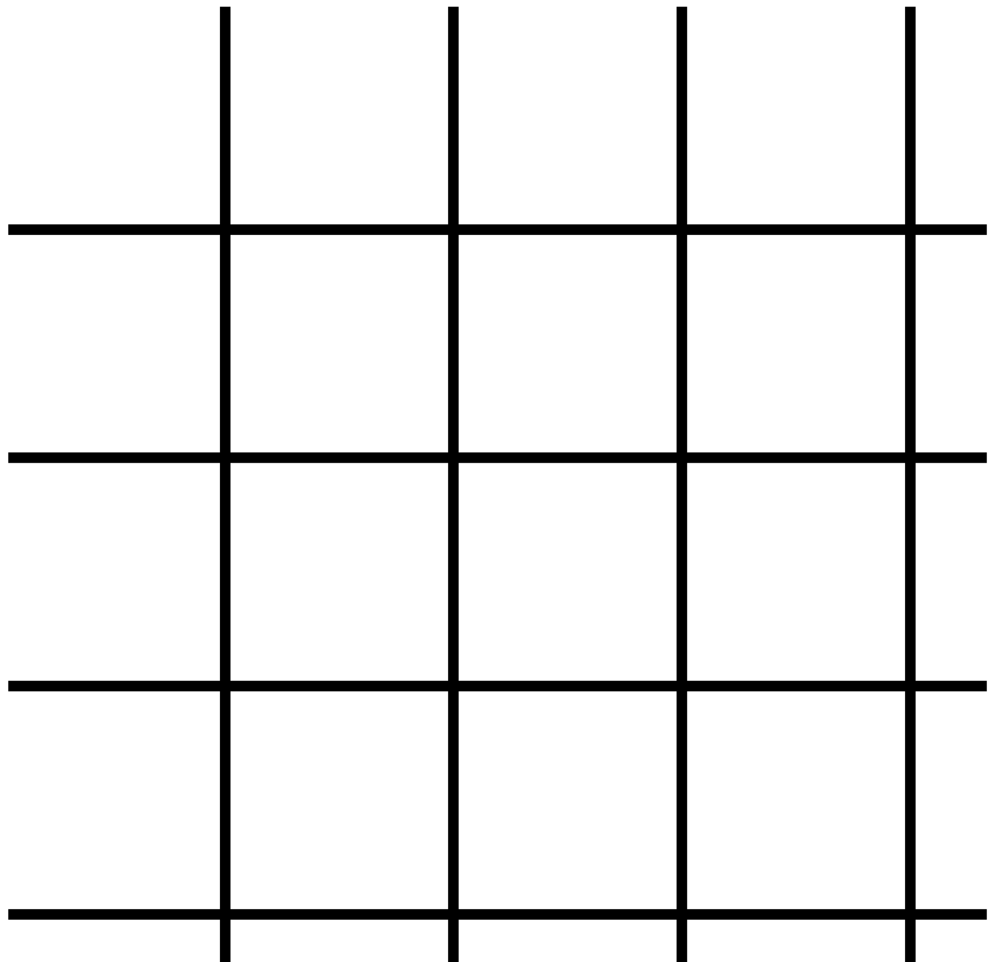
Making new tensor by contraction

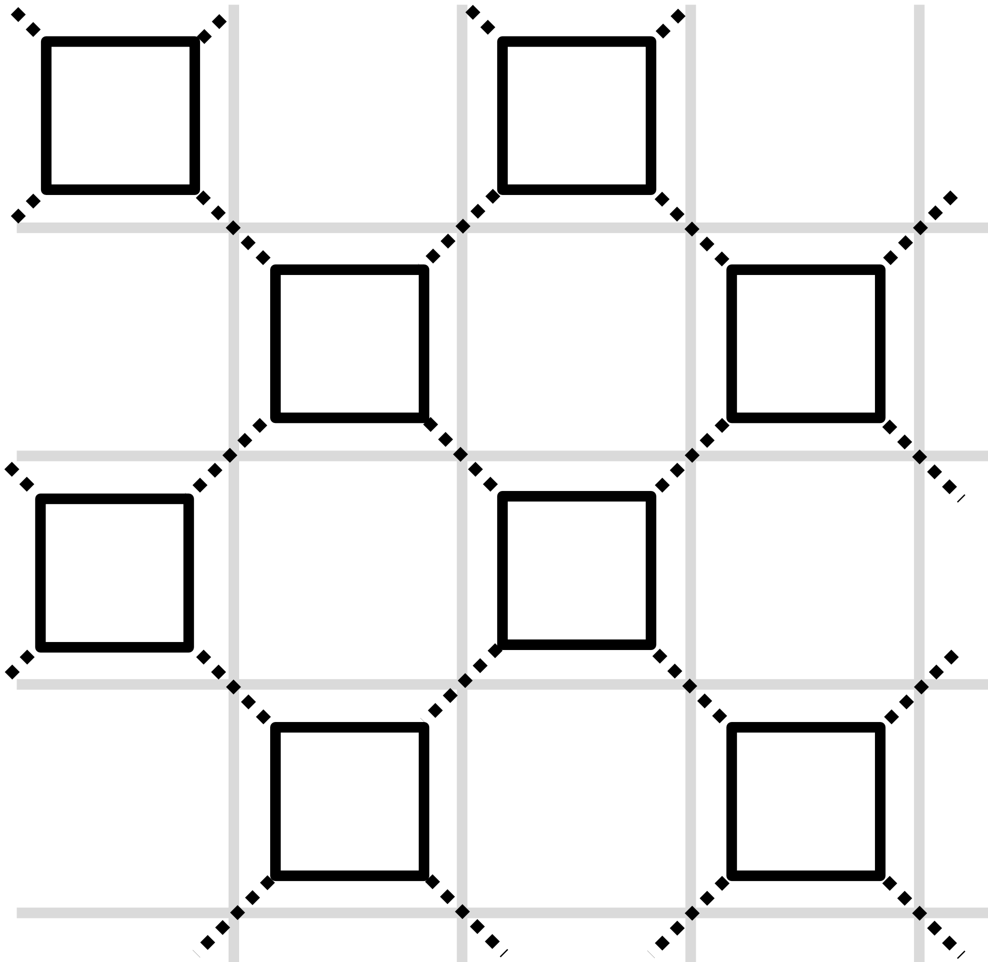
$$\sum_{i,j,k,l}^{\text{all}} (S_1)_{ila}(S_2)_{jib}(S_3)_{kjc}(S_4)_{lkd} = T^{\text{new}}_{abcd}$$

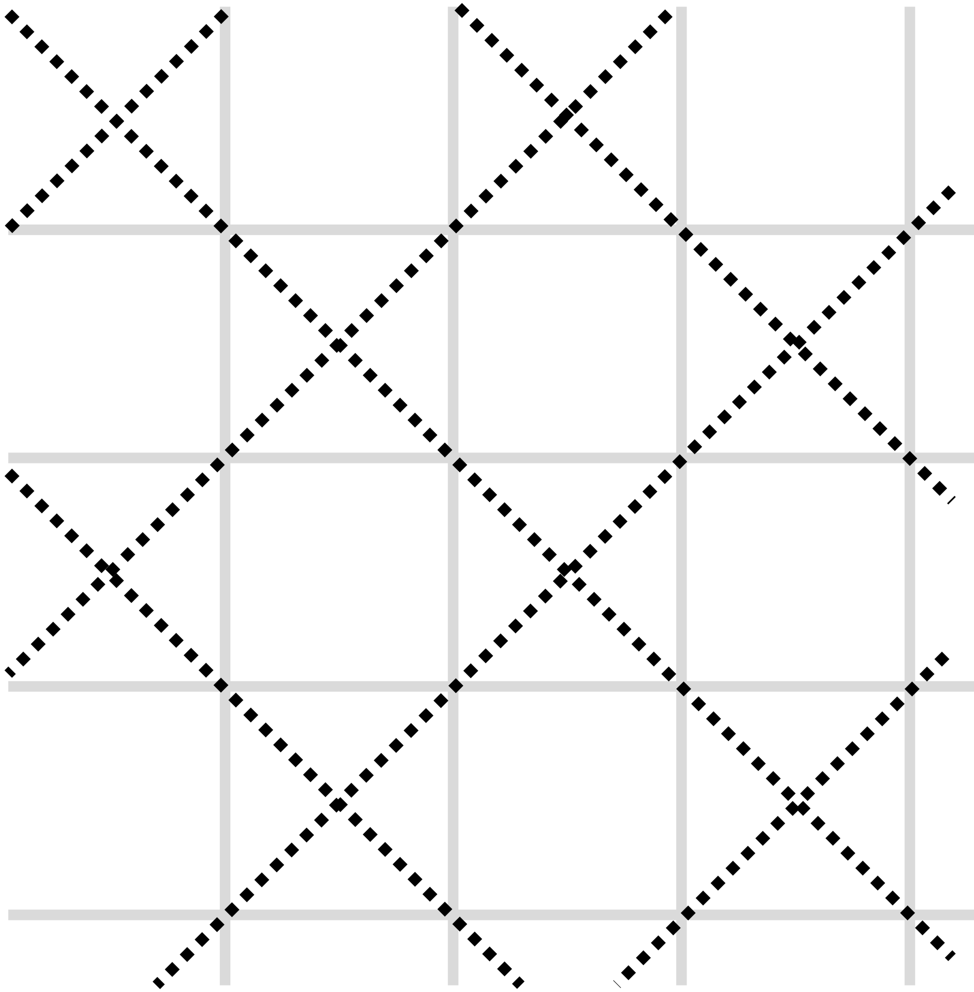
integrate out old d.o.f.

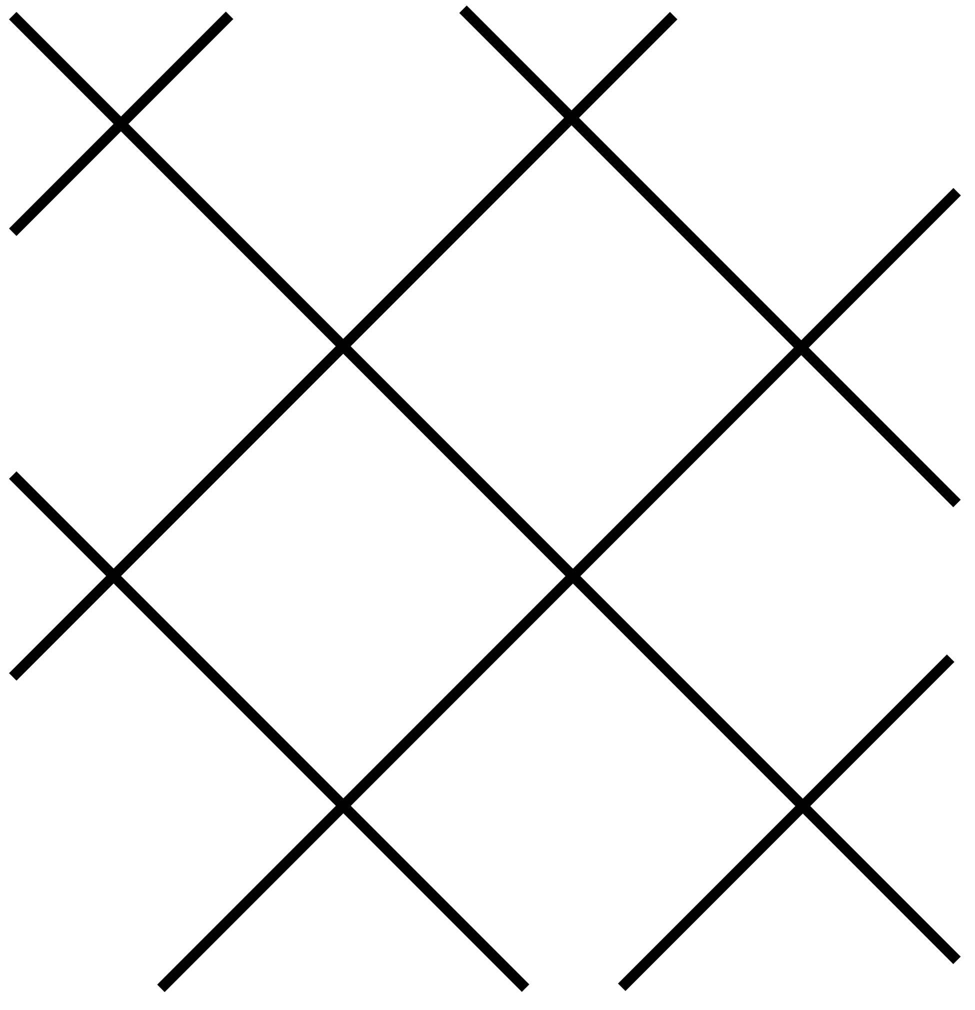
Renormalization-like!

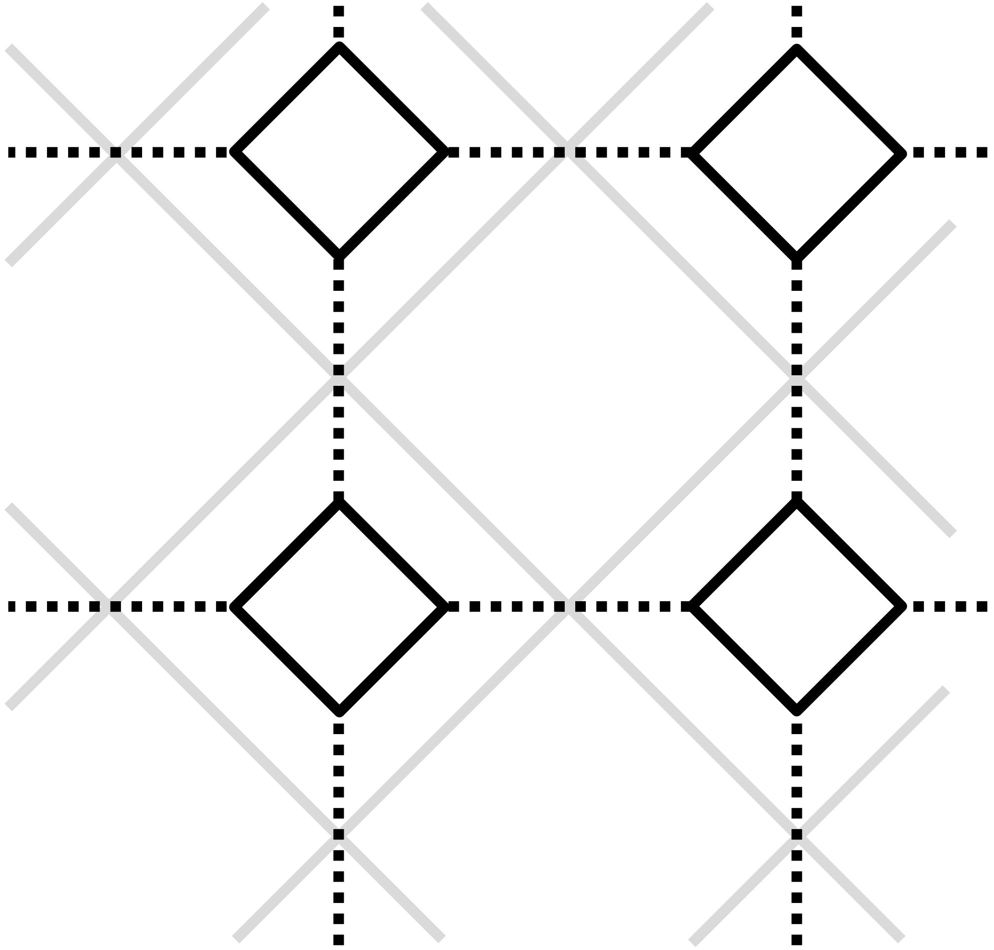


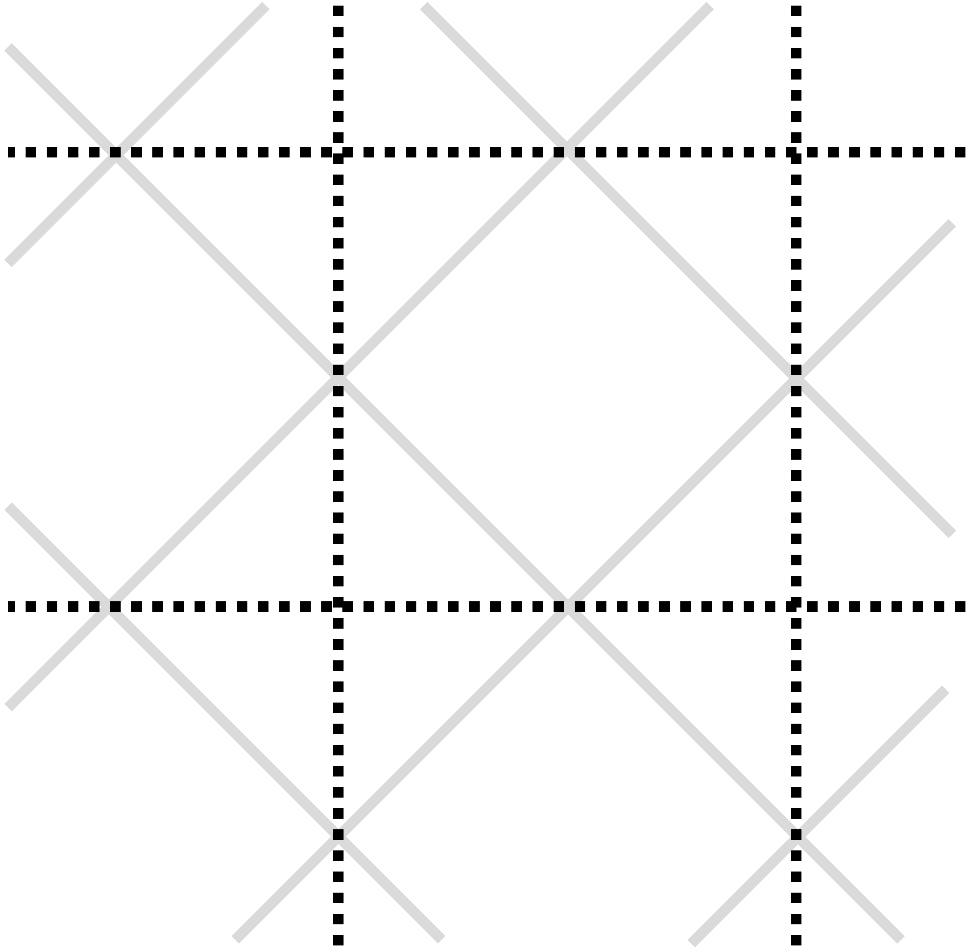


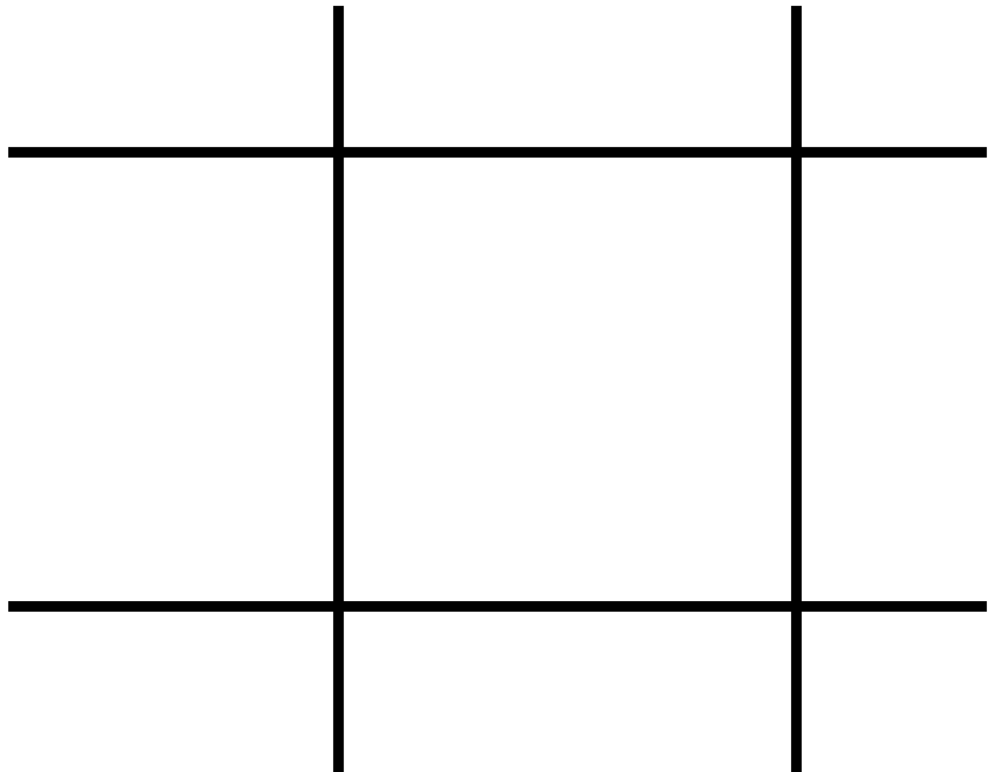


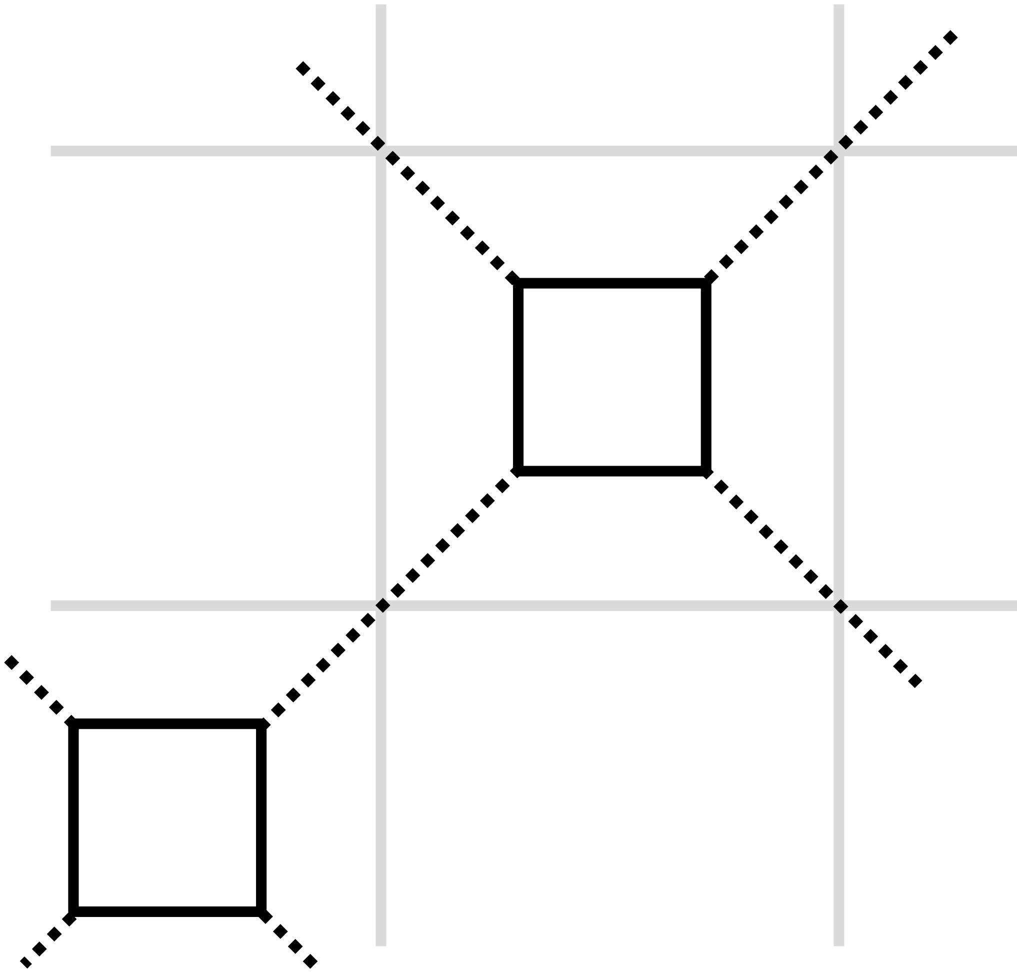


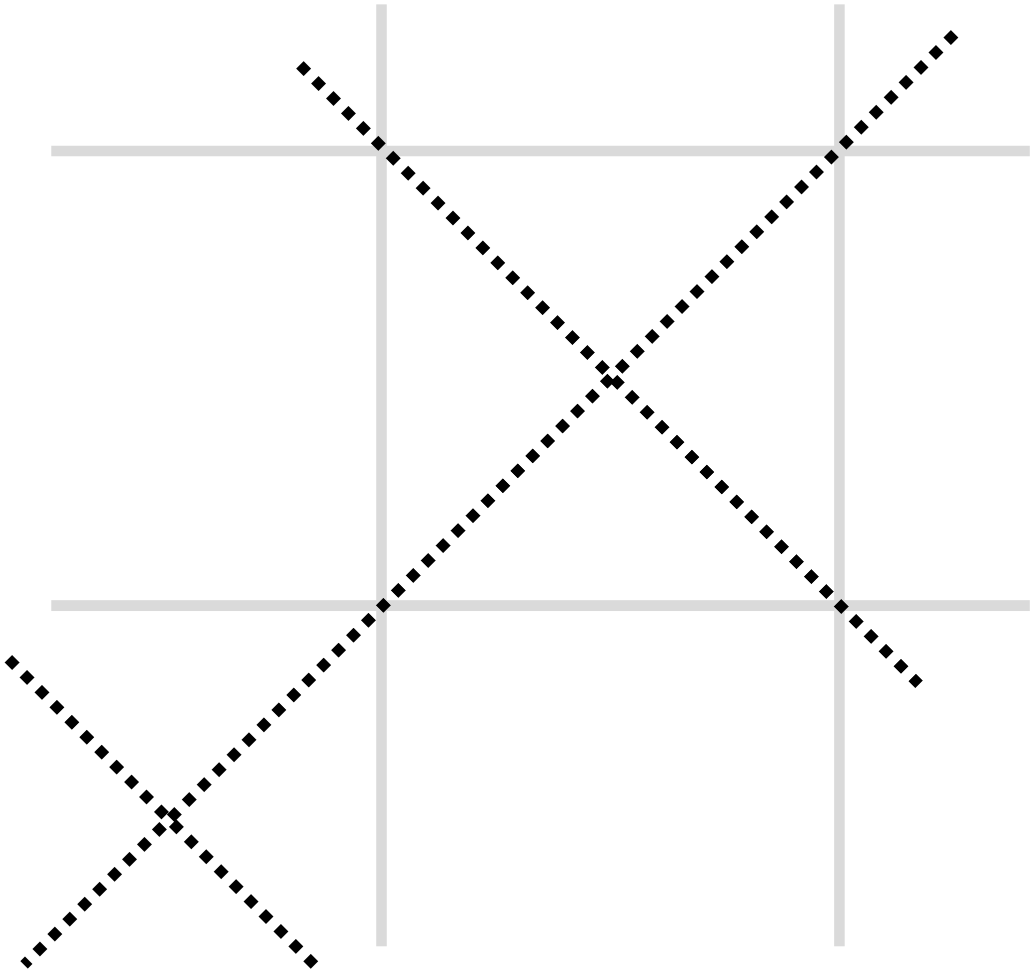


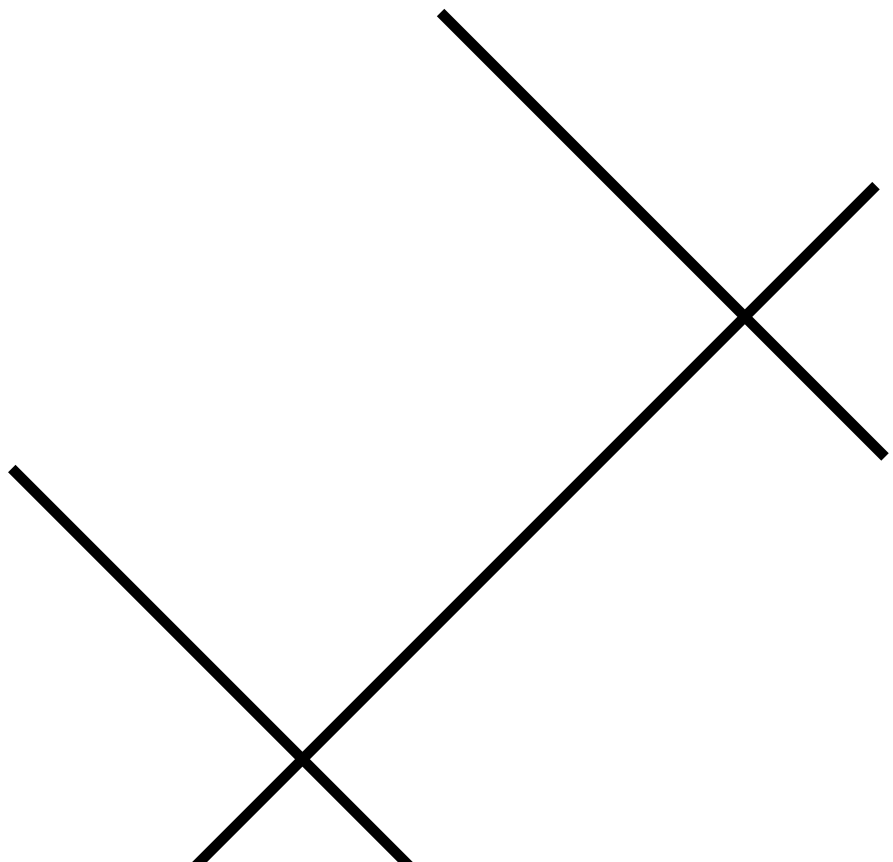


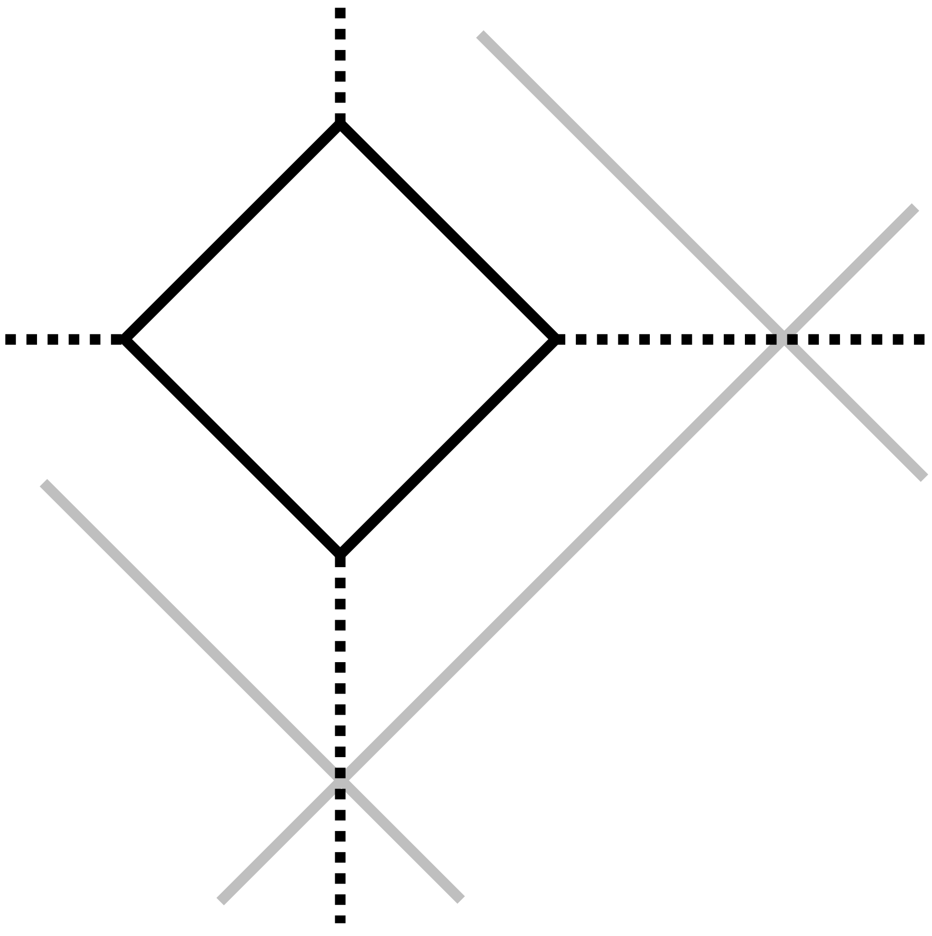


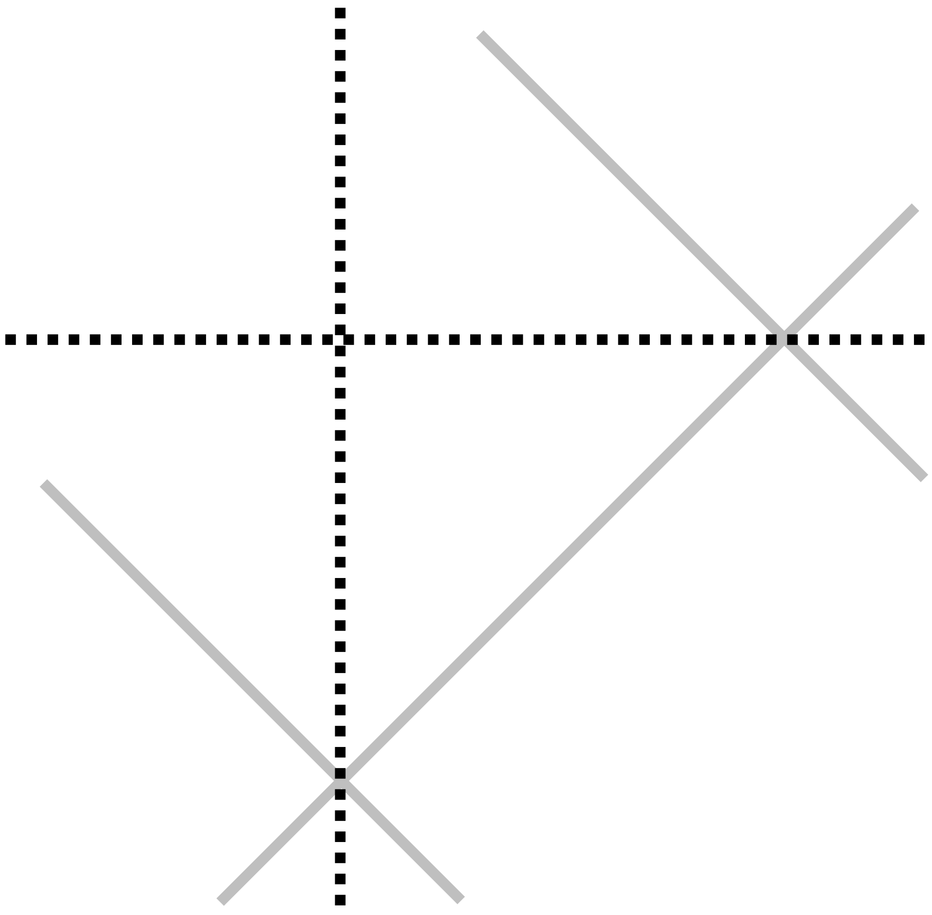


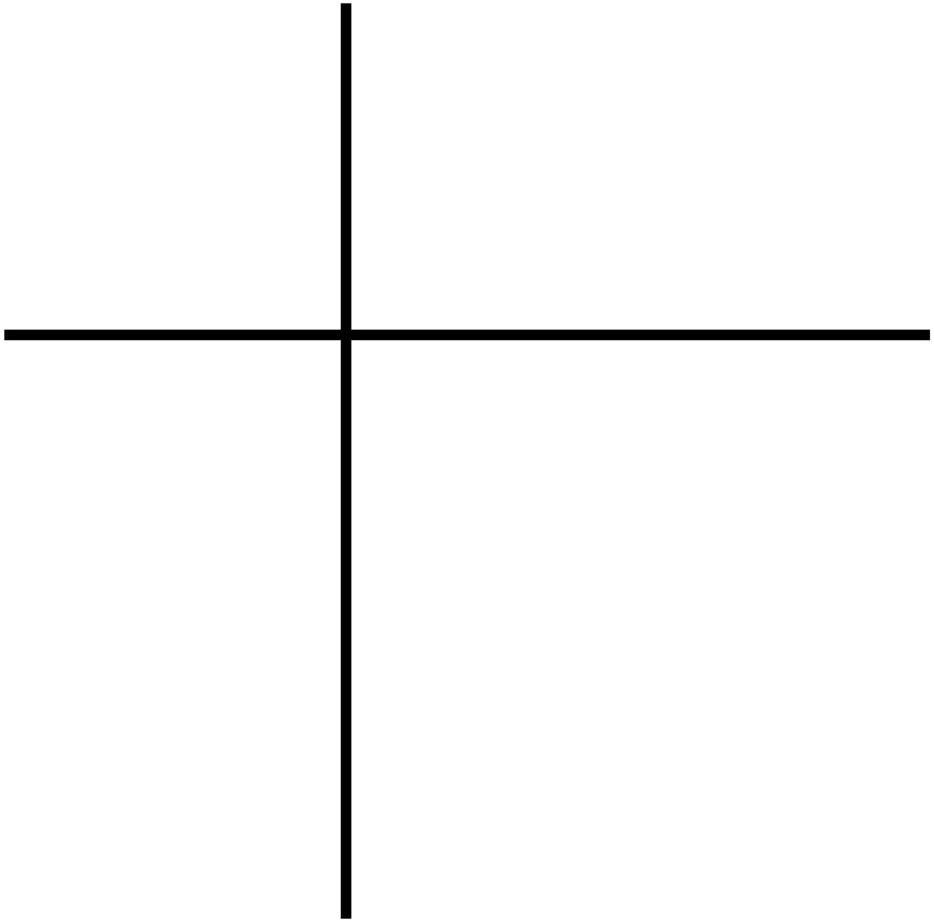


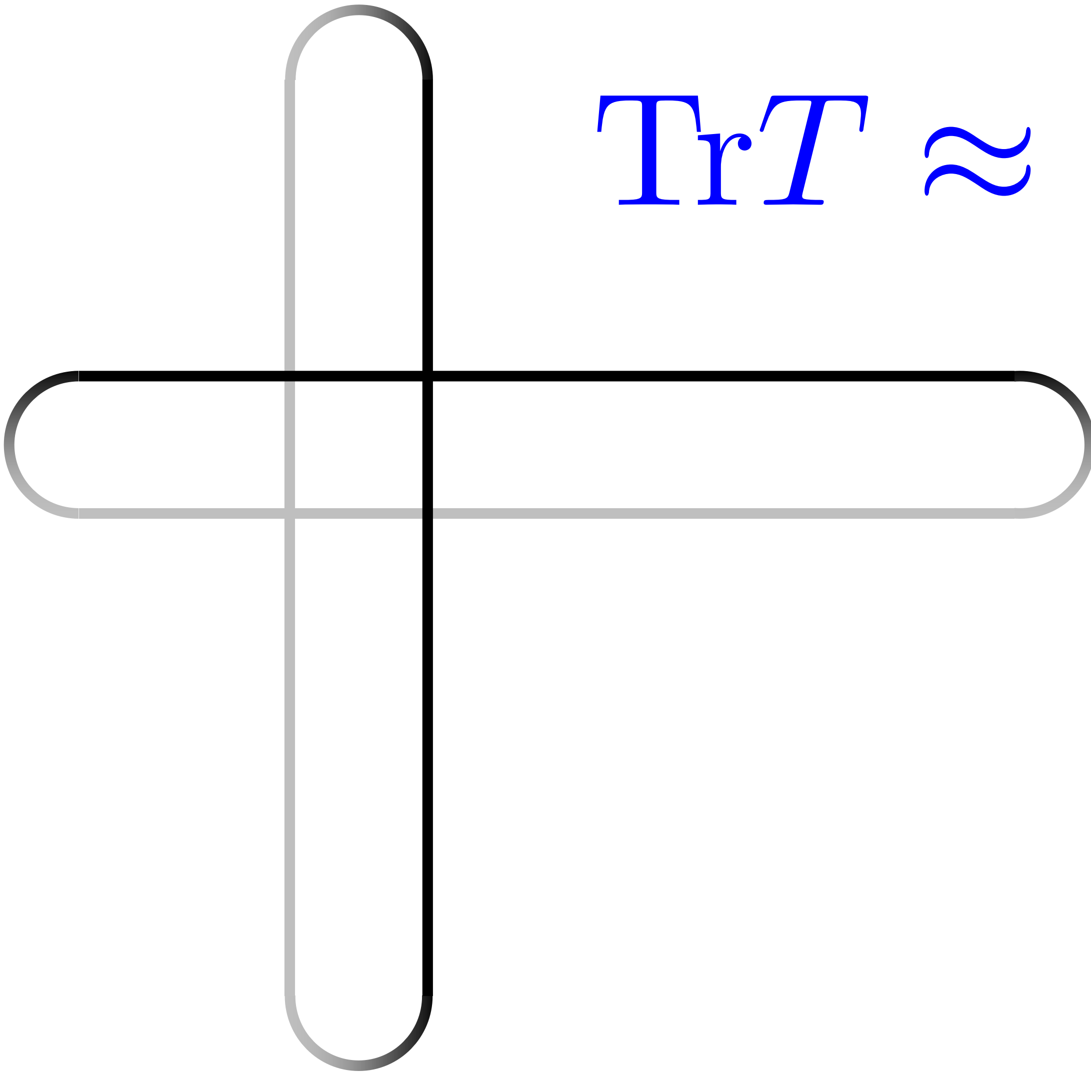






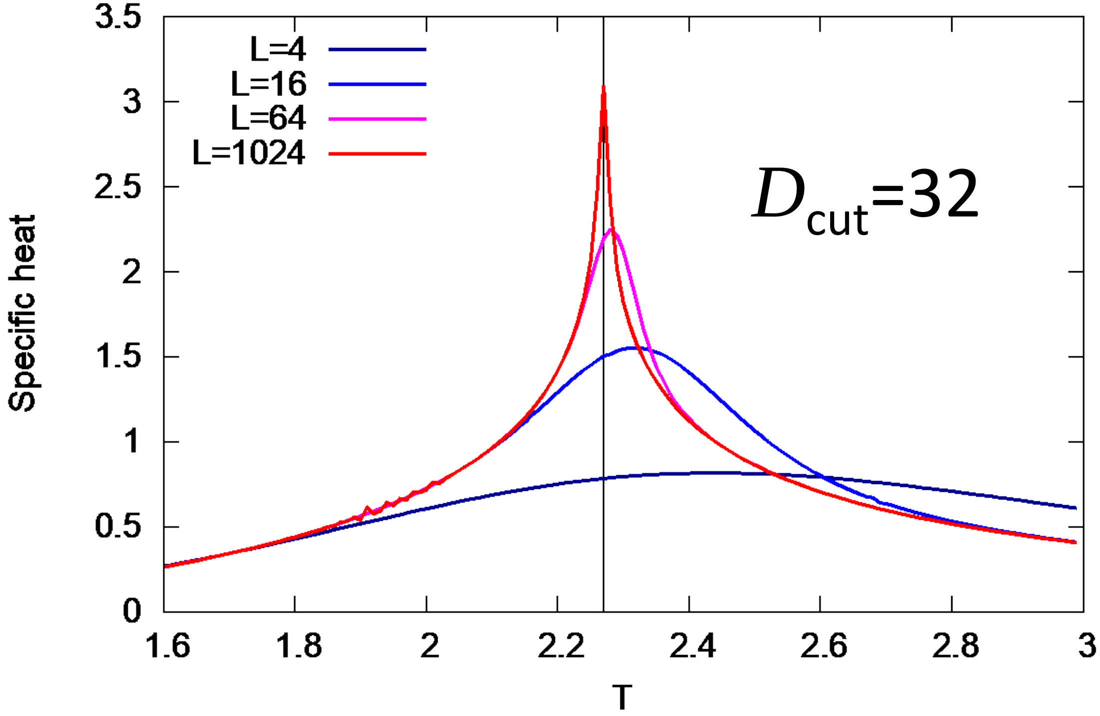






$$\mathrm{Tr}T \approx Z$$

2D Ising model on square lattice



$$C = \frac{1}{L^2}\frac{\partial}{\partial T}\left(T^2 \frac{\partial \ln \mathcal{Z}}{\partial T}\right)$$

numerical derivative

$$T_c = 2/[\ln(1+\sqrt{2})] = 2.269...$$

only one day use of this MBA

Cost ∝ log(Lattice volume) × $(D_{cut})^6$ × [# temperature mesh]

Monte Carlo	Tensor Network
Boltzmann weight is interpreted as probability	Tensor network rep. of partition function (no probability interpretation)
Importance sampling	Information compression by SVD (TRG), Optimization
Statistical errors	Systematic errors (truncated SVD)
Sign problem may appear	No sign problem ∵ no probability
Critical slowing down	Efficiency of compression gets worse around criticality

↓

can be improved by TNR, Loop-TNR in 2D system

Evenbly & Vidal 2014, Gu et al., 2015

Works related with HEP (Lagrangian approach)

- 2D system
  - Spin model : Ising model Levin & Nave PRL99,120601(2007), Aoki et al. Int. Jour. Mod. Phys. B23,18(2009) , X-Y model Meurice et al. PRE89,013308(2014), X-Y model with Fisher zero Meurice et al. PRD89,016008(2014), O(3) model Unmuth-Yockey et al. LATTICE2014, X-Y model + μ Meurice et al. PRE93,012138(2016)
  - Abelian-Higgs Bazavov et al. LATTICE2015
  - ϕ^4 theory Shimizu Mod.Phys.Lett.A27,1250035(2012), Sakai et al., arXiv:1812.00166
  - QED_2 Shimizu & Kuramashi PRD90,014508(2014) & PRD90,034502(2018)
  - $QED_2 + \theta$ Shimizu & Kuramashi PRD90,074503(2014)
  - Gross-Neveu model + μ ST & Yoshimura PTEP043B01(2015)
  - CP(N-1) + θ Kawauchi & ST PRD93,114503(2016)
  - Towards Quantum simulation of O(2) model Zou et al, PRA90,063603
  - N=1 Wess-Zumino model (SUSY model) Sakai et al., JHEP03(2018)141
- 3D system Higher order TRG(HOTRG) : Xie et al. PRB86,045139(2012)
  - 3D Ising, Potts model Wan et al. CPL31,070503(2014)
  - 3D Fermion system Sakai et al.,PTEP063B07(2017)

Tensor network representation for real-time path integral

e.g. 1+1 lattice scalar field theory
with Minkowskian metric

Study of real-time dynamics

- Complex Langevin
  - Real-time correlator, 3+1d φ^4 theory PRL95,202003(2005) Berges et al.
  - (tilted) Schwinger-Keldysh, non-equilibirium, 3+1d SU(2) gauge theory PRD75,045007(2007) Berges et al.
  - convergence issue (difficult for $t >> \beta$)

- Algorithm inspired by Lefschetz thimble PRD95,114501(2017) Alexandru et al.
  - SK setup, 1+1d φ^4 theory
  - Small box (2x8+2)x8 (Larger time extent is harder)

- Tensor network (Here!)

Minkowskian 1+1d Scalar field theory

$$S = \int d^2x \left[\frac{1}{2} \underline{(\partial_\mu \phi)^2} - V(\phi) \right]$$

$\rightarrow$ $(\partial_0 \phi)^2 - (\partial_1 \phi)^2$

$$x = (x_0, x_1)$$

$$\phi \in \mathbb{R}$$

$$\eta_{\mu\nu} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

"purely" Minkowskian but not SK

Minkowskian 1+1d Scalar field theory

$$S = \int d^2x \left[\frac{1}{2} (\partial_\mu \phi)^2 - V(\phi) \right]$$

$$(\partial_\mu \phi)^2 \rightarrow (\partial_0 \phi)^2 - (\partial_1 \phi)^2$$

On lattice

$a = 1$ lattice units

$$\partial_\mu \phi(x) \longrightarrow \phi_{x+\hat{\mu}} - \phi_x$$

$$\int d^2x \longrightarrow \sum_{x \in \mathbb{Z}^2}$$

Path integral

$$Z = \int [d\phi] \exp[iS]$$

Goal: rewrite PI in terms of tensor network

Path integral

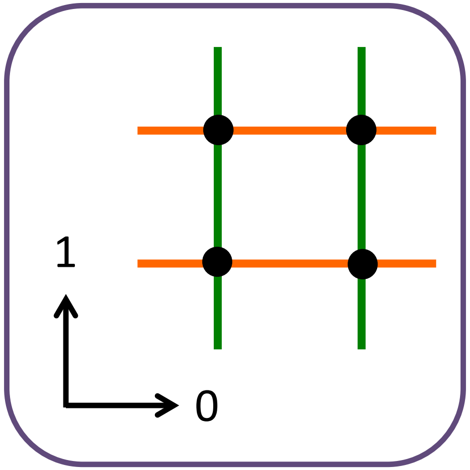
$$Z = \int [d\phi] \exp[iS]$$

$$= \int [d\phi] \prod_x H_0(\phi_x, \phi_{x+\hat{0}}) H_1(\phi_x, \phi_{x+\hat{1}})$$

Goal: rewrite PI in terms of tensor network

$$H_0(\phi, \phi') = \exp\left[+ \frac{i}{2}(\phi - \phi')^2 - \frac{i}{4}V(\phi) - \frac{i}{4}V(\phi') \right]$$

$$H_1(\phi, \phi') = \exp\left[- \frac{i}{2}(\phi - \phi')^2 - \frac{i}{4}V(\phi) - \frac{i}{4}V(\phi') \right]$$



Expansion of H

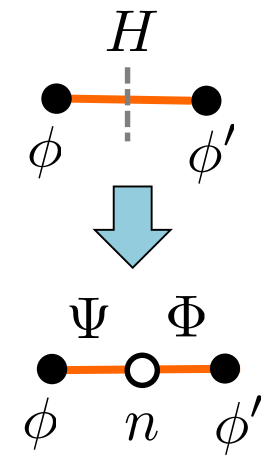
Lay 2002,
Shimizu 2012

For two-variable function

orthonormal basis

$$H_0(\phi,\phi') = \sum_{n=0}^{\infty} \Psi_n^{(0)}(\phi)\ \lambda_n^{(0)}\ \Phi_n^{(0)*}(\phi')$$

singular values



Expansion of H

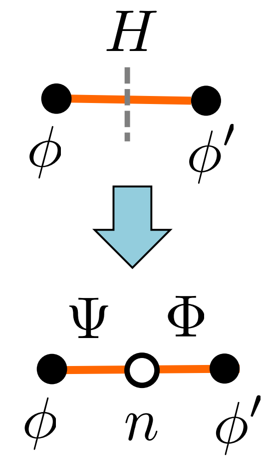
Lay 2002,
Shimizu 2012

For two-variable function

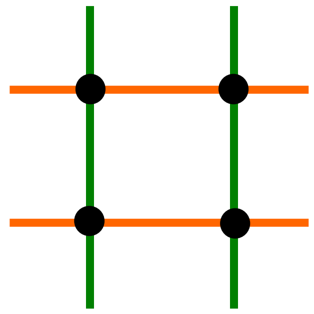
$$H_0(\phi, \phi') = \sum_{n=0}^{\infty} \Psi_n^{(0)}(\phi)\ \lambda_n^{(0)}\ \Phi_n^{(0)*}(\phi')$$

orthonormal basis

singular values

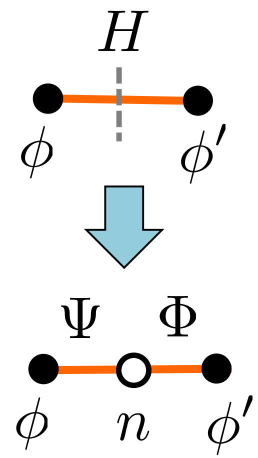


IF orthonormal basis and singular values are obtained,
then tensor is formed as



Expansion of H

Lay 2002,
Shimizu 2012



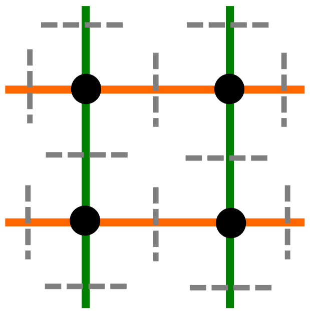
For two-variable function

orthonormal basis

$$H_0(\phi, \phi') = \sum_{n=0}^{\infty} \Psi_n^{(0)}(\phi)\ \lambda_n^{(0)}\ \Phi_n^{(0)*}(\phi')$$

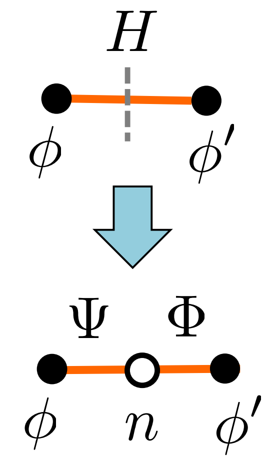
singular values

IF orthonormal basis and singular values are obtained,
then tensor is formed as



Expansion of H

Lay 2002,
Shimizu 2012



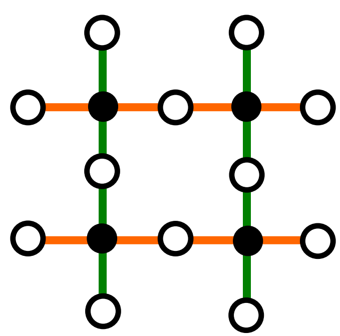
For two-variable function

orthonormal basis

$$H_0(\phi, \phi') = \sum_{n=0}^{\infty} \Psi_n^{(0)}(\phi)\ \lambda_n^{(0)}\ \Phi_n^{(0)*}(\phi')$$

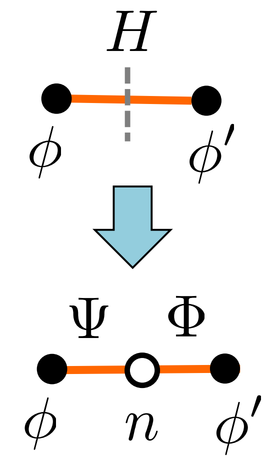
singular values

IF orthonormal basis and singular values are obtained,
then tensor is formed as



Expansion of H

Lay 2002,
Shimizu 2012



For two-variable function

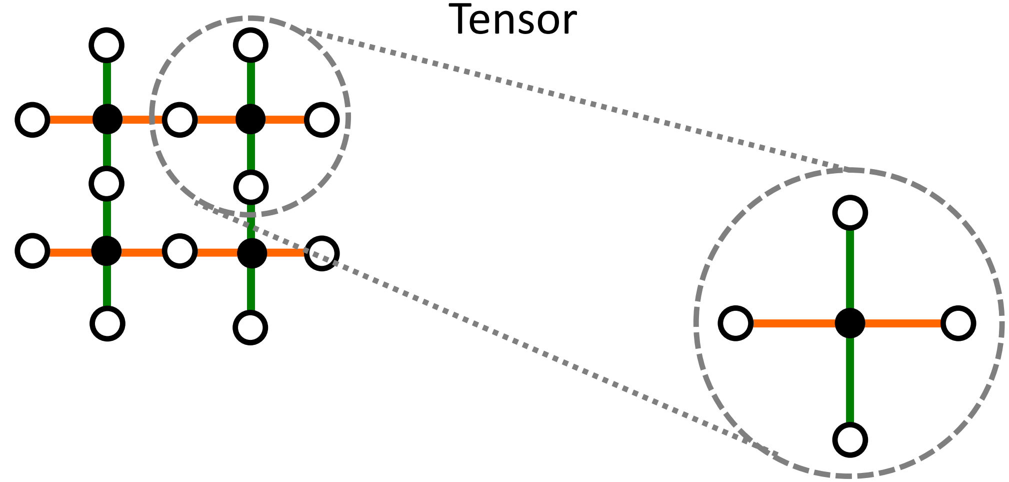
orthonormal basis

$$H_0(\phi, \phi') = \sum_{n=0}^{\infty} \Psi_n^{(0)}(\phi)\ \lambda_n^{(0)}\ \Phi_n^{(0)*}(\phi')$$

singular values

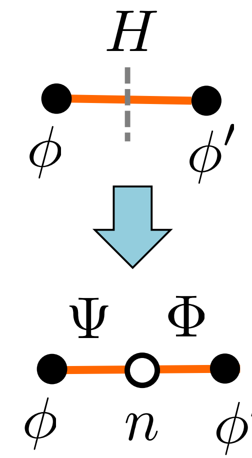
IF orthonormal basis and singular values are obtained,
then tensor is formed as

Tensor



Expansion of H

Lay 2002,
Shimizu 2012



For two-variable function

orthonormal basis

$$H_0(\phi,\phi') = \sum_{n=0}^{\infty} \Psi_n^{(0)}(\phi)\ \lambda_n^{(0)}\ \Phi_n^{(0)*}(\phi')$$

singular values

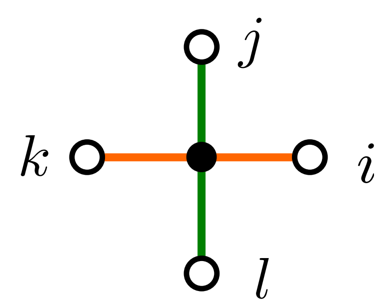
IF orthonormal basis and singular values are obtained,
then tensor is formed as

$$T_{ijkl} = \sqrt{\lambda_i^{(0)}\lambda_j^{(1)}\lambda_k^{(0)}\lambda_l^{(1)}} \int_{-\infty}^{\infty} d\phi \Psi_i^{(0)}\Psi_j^{(1)}\Phi_k^{(0)*}\Phi_l^{(1)*}$$

IF SV has a clear hierarchy $\lambda_0 > \lambda_1 > \lambda_2 > ... \geq 0$

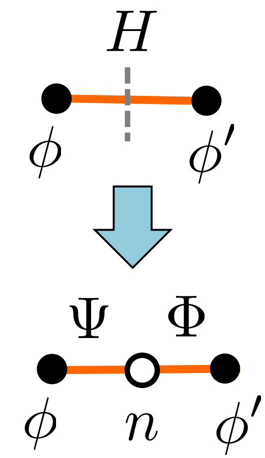
$\Longrightarrow$ truncation is OK $\Longrightarrow$ TRG

$0 \leq i, j, k, l \leq N$



Expansion of H

Lay 2002,
Shimizu 2012



For two-variable function

orthonormal basis

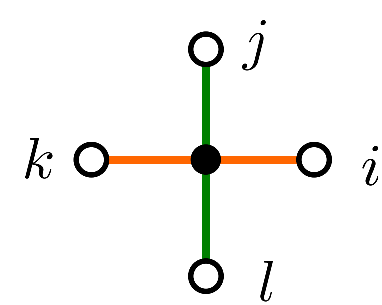
$$H_0(\phi, \phi') = \sum_{n=0}^{\infty} \Psi_n^{(0)}(\phi)\ \lambda_n^{(0)}\ \Phi_n^{(0)*}(\phi')$$

singular values

IF orthonormal basis and singular values are obtained,
then tensor is formed as

$$T_{ijkl} = \sqrt{\lambda_i^{(0)}\lambda_j^{(1)}\lambda_k^{(0)}\lambda_l^{(1)}} \int_{-\infty}^{\infty} d\phi \Psi_i^{(0)} \Psi_j^{(1)} \Phi_k^{(0)*} \Phi_l^{(1)*}$$

Question: How to obtain? Ψ Φ λ



How to obtain Ψ Φ λ

For two-variable function

orthonormal basis

$$H_0(\phi,\phi') = \sum_{n=0}^{\infty} \Psi_n^{(0)}(\phi)\ \lambda_n^{(0)}\ \Phi_n^{(0)*}(\phi')$$

singular values

This expansion holds when H_0 is a compact operator $\int |H_0(x,y)|^2 dxdy < \infty$

$$H_0(\phi,\phi') = \exp\left[\frac{i}{2}(\phi-\phi')^2 - \frac{i}{4}V(\phi) - \frac{i}{4}V(\phi')\right]$$

How to obtain $\Psi \ \ \Phi \ \ \lambda$

For two-variable function

orthonormal basis

$$H_0(\phi,\phi') = \sum_{n=0}^{\infty} \Psi_n^{(0)}(\phi)\ \lambda_n^{(0)}\ \Phi_n^{(0)*}(\phi')$$

singular values

This expansion holds when H_0 is a compact operator $\int |H_0(x,y)|^2 dx dy < \infty$

$$H_0(\phi,\phi') = \exp\left[\frac{i}{2}(\phi-\phi')^2 - \frac{i}{4}V(\phi) - \frac{i}{4}V(\phi')\right]$$

For free $V = \frac{m_0^2}{2}\phi^2$ and complex mass $m_0^2 \longrightarrow m_0^2 - i\epsilon \ \ (\epsilon > 0)$

$$= e^{\frac{i}{2}\phi^2} \exp\left[-i\phi\phi' + (-im_0^2 - \epsilon)\frac{\phi^2+\phi'^2}{8}\right] e^{\frac{i}{2}\phi'^2}$$

damping factor → compact

cancel when making tensor

How to obtain $\Psi \quad \Phi \quad \lambda$

$$\exp\left[-i\phi\phi' + (-im_0^2 - \epsilon)\frac{\phi^2 + \phi'^2}{8}\right]$$

up to 2π factor

Remember 1-dim QM

$$e^{ixp} = \langle x|p\rangle = \sum_{n=0}^{\infty}\langle x|n\rangle\langle n|p\rangle$$

$$= \sum_{n=0}^{\infty}\psi_n(x)\tilde{\psi}_n^*(p)$$

How to obtain $\Psi \;\; \Phi \;\; \lambda$

$$\exp\left[-i\phi\phi' + (-im_0^2 - \epsilon)\frac{\phi^2 + \phi'^2}{8}\right]$$

up to 2π factor

Remember 1-dim QM

$$e^{ixp} = \langle x|p\rangle = \sum_{n=0}^{\infty}\langle x|n\rangle\langle n|p\rangle$$

$$\tilde{\psi}_n^*(p) = i^n \psi_n^*(p)$$

$$= \sum_{n=0}^{\infty}\psi_n(x)\tilde{\psi}_n^*(p)$$

if basis is Hermite function

$$\psi_n(x) = \frac{1}{\sqrt{\pi^{1/2} n! 2^n}} H_n(x) e^{-x^2/2}$$

$$= \sum_{n=0}^{\infty}\psi_n(x)\; i^n \;\psi_n(p)$$

Truncation is not allowed

How to obtain $\Psi \quad \Phi \quad \lambda$

$$\exp\left[-i\phi\phi' + (-im_0^2 - \epsilon)\frac{\phi^2 + \phi'^2}{8}\right]$$

$\mathrm{Re}[\beta] > 0$

up to 2π factor

$$e^{ixp}\underline{e^{-\beta(x^2+p^2)}} = \sum_{n=0}^{\infty} \left(\underline{e^{-\beta x^2}\psi_n(x)}\right) \; i^n \; \left(e^{-\beta p^2}\psi_n(p)\right)$$

damping factor

$$\sum_{m=0}^{\infty} \underline{G_{nm}}\psi_m(x)$$

$$G_{nm} = \int_{-\infty}^{\infty} dx e^{-\beta x^2}\psi_m(x)\psi_n(x)$$

How to obtain Ψ Φ λ

$$\exp\left[-i\phi\phi' + (-im_0^2 - \epsilon)\frac{\phi^2 + \phi'^2}{8}\right]$$

$\mathrm{Re}[\beta] > 0$

up to 2π factor

$$e^{ixp}e^{-\beta(x^2+p^2)} = \sum_{n=0}^{\infty}\left(e^{-\beta x^2}\psi_n(x)\right)\; i^n\; \left(e^{-\beta p^2}\psi_n(p)\right)$$

$$= \sum_{n=0}^{\infty}\left(\sum_{m=0}^{\infty} G_{nm}\psi_m(x)\right) i^n \left(\sum_{k=0}^{\infty} G_{nk}\psi_k(p)\right)$$

How to obtain $\Psi \ \ \Phi \ \ \lambda$

$$\exp\left[-i\phi\phi' + (-im_0^2 - \epsilon)\frac{\phi^2 + \phi'^2}{8}\right]$$

$\mathrm{Re}[\beta] > 0$

up to 2π factor

$$e^{ixp}e^{-\beta(x^2+p^2)} = \sum_{n=0}^{\infty}\left(e^{-\beta x^2}\psi_n(x)\right)\ i^n\ \left(e^{-\beta p^2}\psi_n(p)\right)$$

$$= \sum_{n=0}^{\infty}\left(\sum_{m=0}^{\infty}G_{nm}\psi_m(x)\right)i^n\left(\sum_{k=0}^{\infty}G_{nk}\psi_k(p)\right)$$

$$= \sum_{m,k=0}^{\infty}\psi_m(x)\left(\sum_{n=0}^{\infty}i^n G_{nm}G_{nk}\right)\psi_k(p)$$

How to obtain $\Psi \;\; \Phi \;\; \lambda$

$$\exp\left[-i\phi\phi' + (-im_0^2 - \epsilon)\frac{\phi^2 + \phi'^2}{8}\right]$$

$\mathrm{Re}[\beta] > 0$

up to 2π factor

$$e^{ixp} e^{-\beta(x^2+p^2)} = \sum_{n=0}^{\infty} \left(e^{-\beta x^2}\psi_n(x)\right) \; i^n \; \left(e^{-\beta p^2}\psi_n(p)\right)$$

$$= \sum_{n=0}^{\infty} \left(\sum_{m=0}^{\infty} G_{nm}\psi_m(x)\right) i^n \left(\sum_{k=0}^{\infty} G_{nk}\psi_k(p)\right)$$

$$= \sum_{m,k=0}^{\infty} \psi_m(x) \left(\sum_{n=0}^{\infty} i^n G_{nm} G_{nk}\right) \psi_k(p)$$

$$X_{mk} = \sum_{a=0}^{\infty} U_{ma}\lambda_a (V^\dagger)_{ak}$$

How to obtain $\Psi \quad \Phi \quad \lambda$

$$\exp\left[-i\phi\phi' + (-im_0^2 - \epsilon)\frac{\phi^2 + \phi'^2}{8}\right]$$

$\mathrm{Re}[\beta] > 0$

up to 2π factor

$$e^{ixp}e^{-\beta(x^2+p^2)} = \sum_{n=0}^{\infty}\left(e^{-\beta x^2}\psi_n(x)\right)\ i^n\ \left(e^{-\beta p^2}\psi_n(p)\right)$$

$$= \sum_{n=0}^{\infty}\left(\sum_{m=0}^{\infty} G_{nm}\psi_m(x)\right) i^n \left(\sum_{k=0}^{\infty} G_{nk}\psi_k(p)\right)$$

$$= \sum_{m,k=0}^{\infty} \psi_m(x)\left(\sum_{n=0}^{\infty} i^n G_{nm}G_{nk}\right)\psi_k(p)$$

$$= \sum_{a=0}^{\infty}\left(\sum_{m=0}^{\infty}\psi_m(x)U_{ma}\right)\lambda_a\left(\sum_{k=0}^{\infty}(V^\dagger)_{ak}\psi_k(p)\right)$$

How to obtain $\Psi \quad \Phi \quad \lambda$

$$\exp\left[-i\phi\phi' + (-im_0^2 - \epsilon)\frac{\phi^2 + \phi'^2}{8}\right]$$

$\mathrm{Re}[\beta] > 0$

up to 2π factor

$$e^{ixp}e^{-\beta(x^2+p^2)} = \sum_{n=0}^{\infty}\left(e^{-\beta x^2}\psi_n(x)\right)\ i^n\ \left(e^{-\beta p^2}\psi_n(p)\right)$$

$$= \sum_{n=0}^{\infty}\left(\sum_{m=0}^{\infty} G_{nm}\psi_m(x)\right) i^n \left(\sum_{k=0}^{\infty} G_{nk}\psi_k(p)\right)$$

$$= \sum_{m,k=0}^{\infty} \psi_m(x)\left(\sum_{n=0}^{\infty} i^n G_{nm}G_{nk}\right)\psi_k(p)$$

$$= \sum_{a=0}^{\infty}\underline{\left(\sum_{m=0}^{\infty}\psi_m(x)U_{ma}\right)}\lambda_a\underline{\left(\sum_{k=0}^{\infty}(V^\dagger)_{ak}\psi_k(p)\right)}$$

$$= \sum_{a=0}^{\infty}\underline{\Psi_a(x)}\lambda_a\underline{\Phi_a^*(p)}$$

How to obtain Ψ Φ λ

$$\exp\left[-i\phi\phi' + (-im_0^2 - \epsilon)\frac{\phi^2 + \phi'^2}{8}\right] = \sum_{a=0}^{\infty} \Psi_a(\phi)\lambda_a\Phi_a^*(\phi')$$

① $$G_{nm} = \int_{-\infty}^{\infty} dx e^{-(im_0^2+\epsilon)x^2/8}\psi_m(x)\psi_n(x)$$

② $$X_{mk} = \sum_{n=0}^{\infty} i^n G_{nm}G_{nk}$$

range of *m*, *n*, *k* is truncated at $K(\gg N)$

③ $$X_{mk} \approx \sum_{a=0}^{N} U_{ma}\lambda_a(V^\dagger)_{ak}$$

(SVD)

truncated at N

→ tensor $\approx N^4$

④ $$\begin{cases} \Psi = U\psi \\ \Phi = V\psi \end{cases}$$

ψ Hermite function

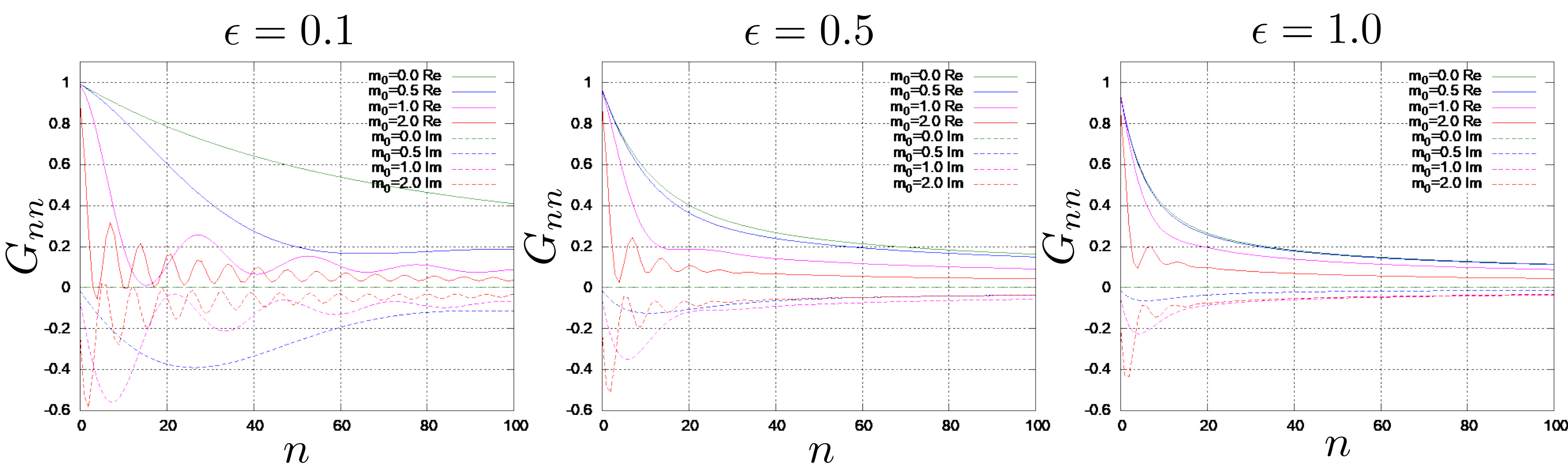
Numerical results

$$G_{nm} = \int_{-\infty}^{\infty} dx e^{-(im_0^2+\epsilon)x^2/8} \psi_m(x)\psi_n(x)$$

no sign problem

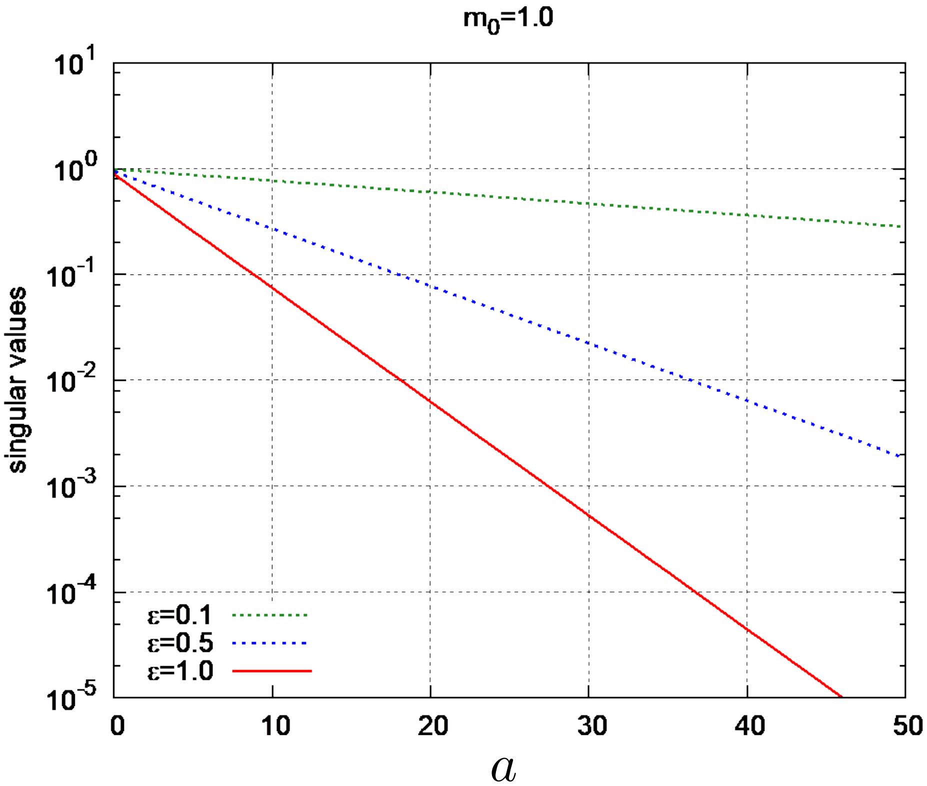
$$\beta = im_0^2 + \epsilon$$

$$G_{m+1,n+1} = \frac{1}{(1+\beta)\sqrt{(m+1)(n+1)}} \Big[G_{mn} + (1-\beta)\sqrt{mn}\, G_{m-1,n-1} - \beta\sqrt{(m+1)n}\, G_{m+1,n-1} - \beta\sqrt{m(n+1)}\, G_{m-1,n+1} \Big] .$$



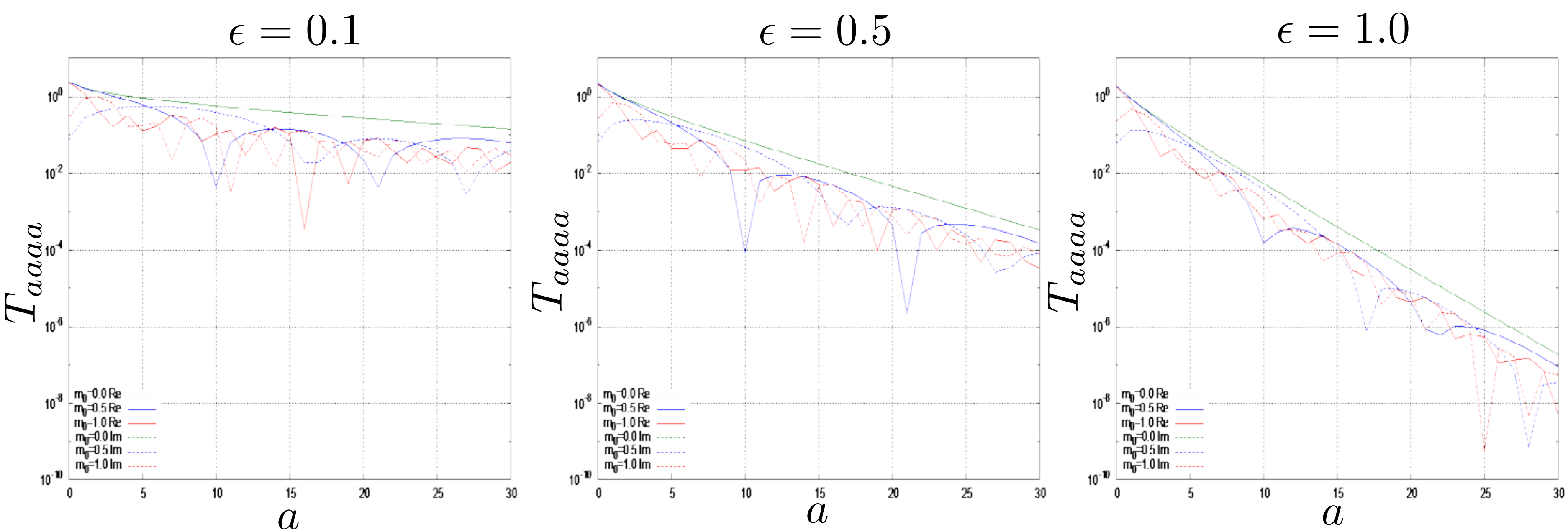
Singular values

$$X_{mk} \equiv \sum_{n=0}^{\infty} i^n G_{nm} G_{nk} = \sum_{a=0}^{\infty} U_{ma} \underline{\lambda_a} V_{ak}^{\dagger}$$



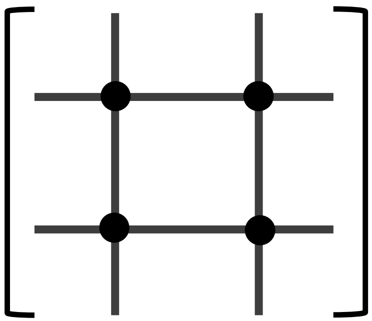
$$T_{abcd} \equiv \sqrt{\lambda_a^{(0)}\lambda_b^{(1)}\lambda_c^{(0)}\lambda_d^{(1)}} \int_{-\infty}^{\infty} d\phi \Psi_a^{(0)} \Psi_b^{(1)} \Phi_c^{(0)*} \Phi_d^{(1)*}$$

The integral can be estimated by a recursion relation



Z on 2x2 lattice of free case

$Z = \mathrm{Tr}$



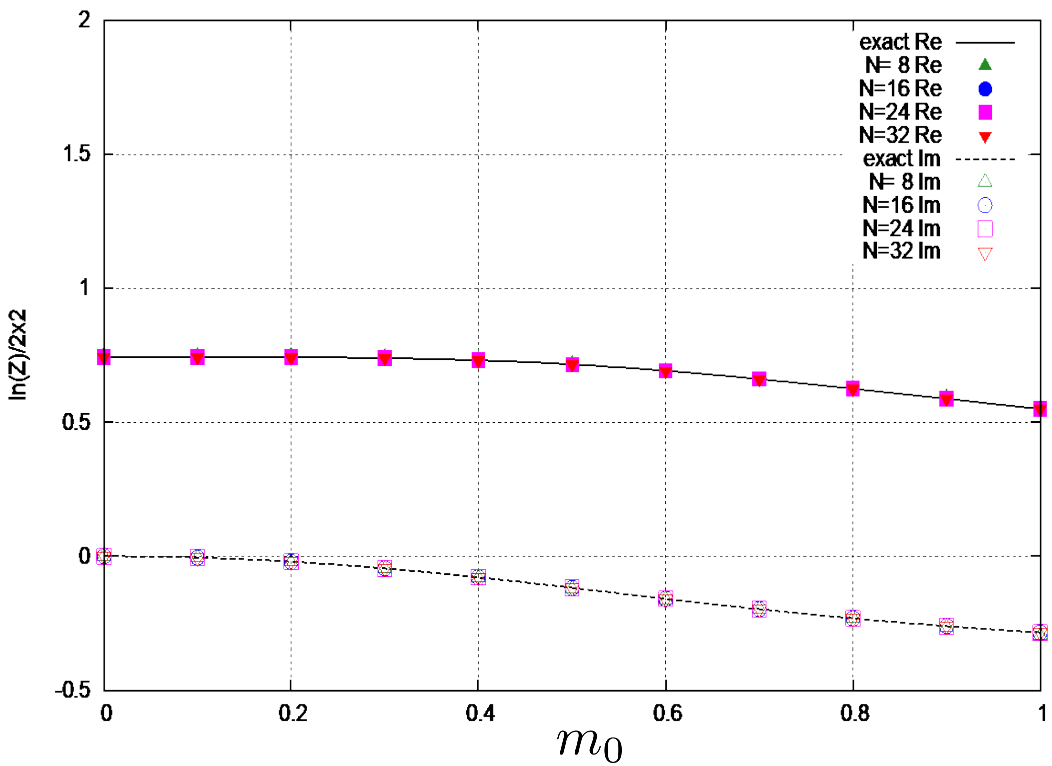
Periodic BC : not physical

T_{abcd}

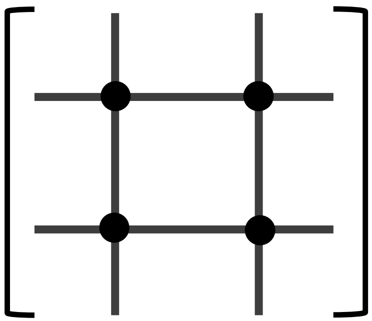
$0 \leq a, b, c, d \leq N$

contraction for 2x2 tensor
(no coarse graining)

$\epsilon = 0.5$



Z on 2x2 lattice of free case

Z = Tr 

Periodic BC : not physical

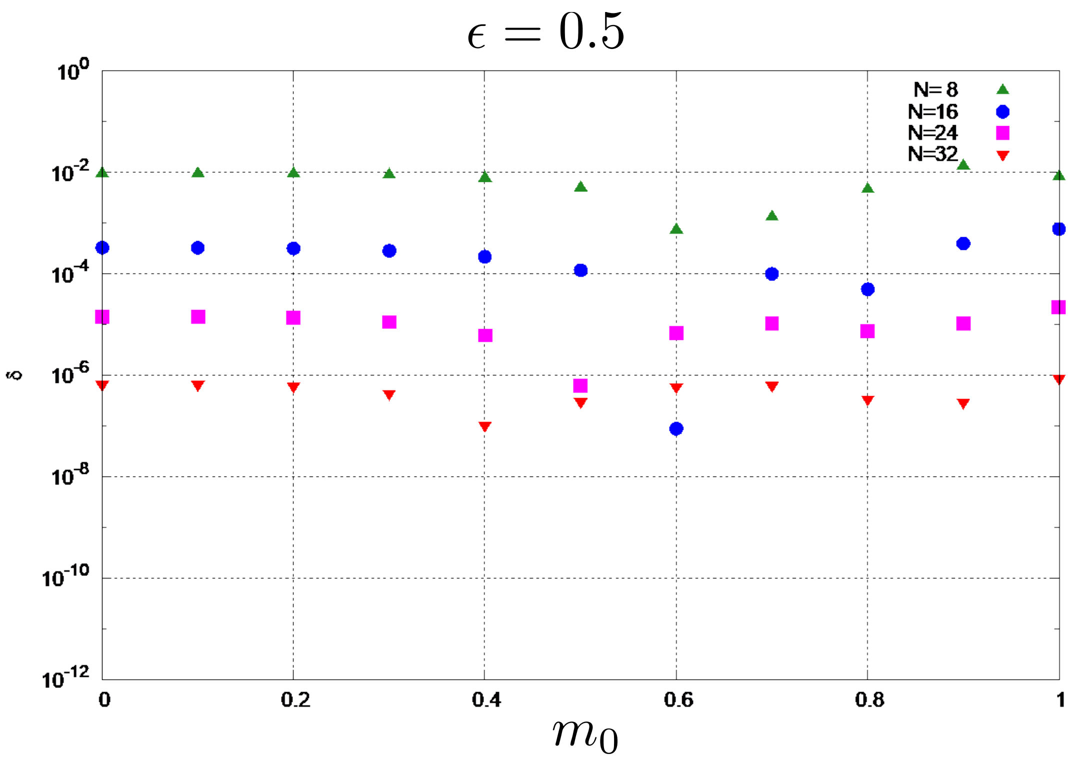
$$T_{abcd}$$

$$0 \le a, b, c, d \le N$$

for real part of Z

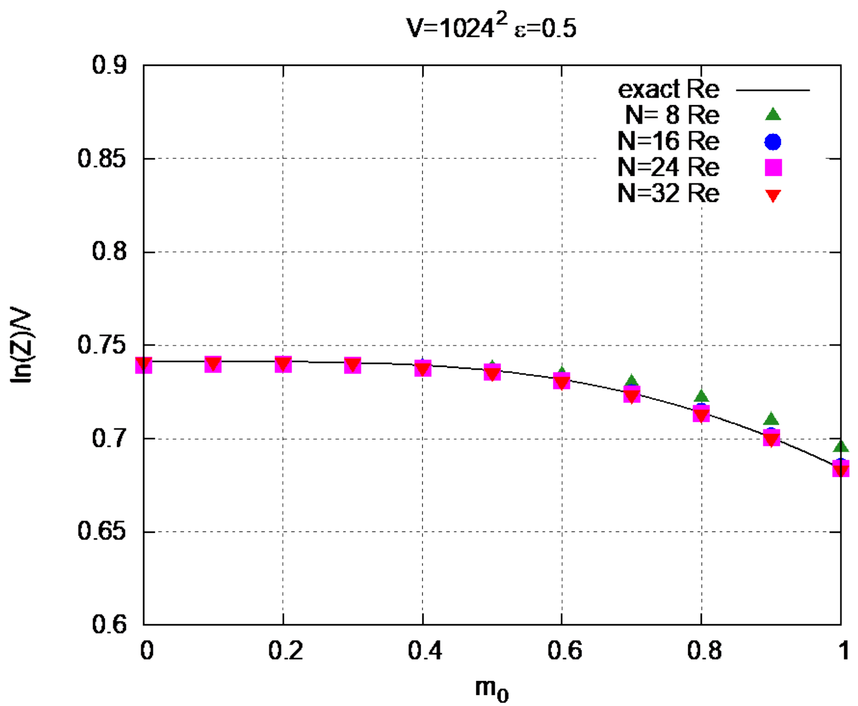
$$\delta = \left| \frac{\ln Z - \ln Z_{\rm exact}}{\ln Z_{\rm exact}} \right|$$

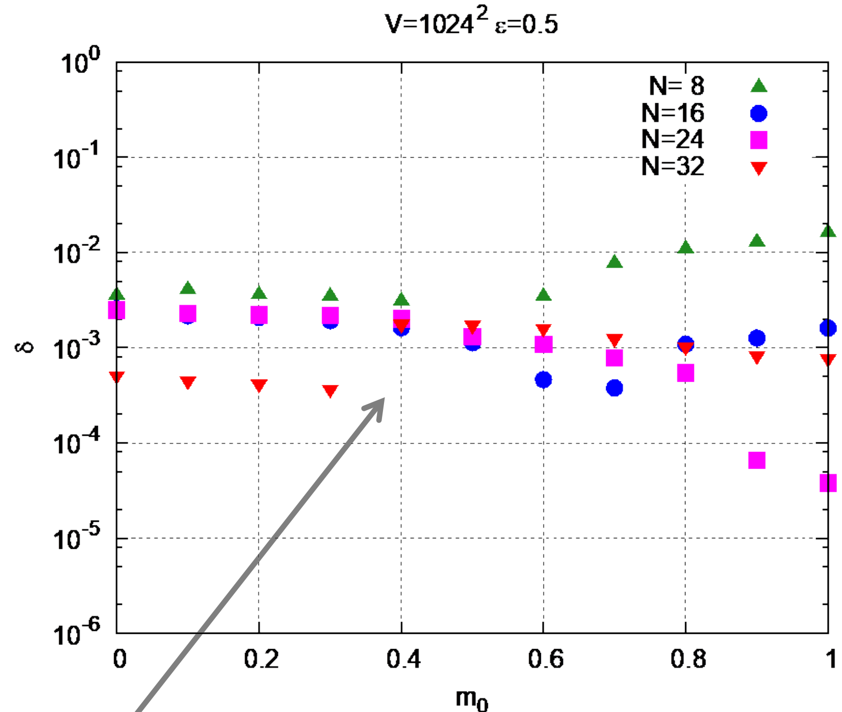
contraction for 2x2 tensor
(no coarse graining)



Larger volume in free case

Using TRG for coarse-graining $V = (1024)^2$ $\epsilon = 0.5$





need improved algorithm: TNR, Loop-TNR, GILT?

Summary

- Tensor network representation for scalar field theory with Minkowskian metric is derived
- Orthonormal basis function (Hermite function) plays an important role (SVD & avoid the sign problem)
- Feynman prescription ($m_0^2 - i\varepsilon$) provides a damping factor
- For 2x2 lattice, it works wide range of mass for free case with ε=0.5
- For larger volume, the precision of Z tends to be worse (need TNR/Loop-TNR/GILT?)
- No sign problem but there is a problem of information compressibility near singular points (hierarchy of singular values)

Future

- Improvement of initial tensor using idea of TNR, GILT, etc
- Tilted time axis $t \longrightarrow te^{-i\xi/2}$ (instead of ${m_0}^2 - i\varepsilon$)
- Interacting case
- Schwinger-Keldysh, Out of equilibrium
- Real-time correlator, Spectral function, Transport coefficients
- Other models including fermions and gauge fields
- Higher dimensional system (Hard!!!)

(Personal) Road map of tensor network approach

- Tensor network representation
  - Scalar, Fermion, Gauge
  - Minkowskian space-time Done!!!
  - Chiral gauge theory ???
- Cost of coarse-graining
  - Higher dimension (MC, optimization) in progress
  - Large # of Internal degree of freedom

    e.g. SU(N) ???

Free Energy

$$\mathcal{Z} = \int [d\phi] e^{iS} = \sqrt{\frac{(-2\pi i)^V}{\prod_p K(p)}}$$

$$K(p) = -(2\sin(p_0/2))^2 + (2\sin(p_1/2))^2 + m_0^2$$

$$p_\mu = \frac{2\pi}{L_\mu} n_\mu, \qquad n_\mu = 0, 1, 2, ..., L_\mu - 1.$$

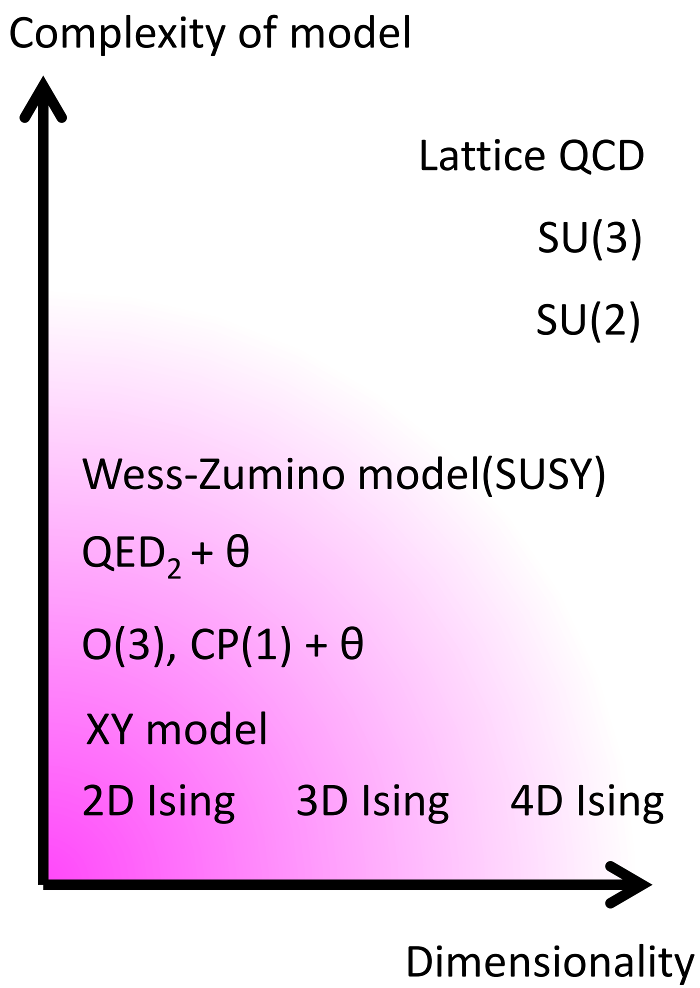
For 2x2 lattice

$$\mathcal{Z}^{(2\times 2)} = \frac{(2\pi)^2}{m_0^2\sqrt{(m_0^2+4)(m_0^2-4)}}.$$

singular points

$$m_0 = 0,\ \pm 2,\ \pm 2i$$

Future



- Further Impr. of HOTRG

  Better coarse-graining in higher dimensional system
- Cost reduction

  MC/Projective truncation method
- Memory & cost reduction using symmetry

  using block structure of tensor
- Parallelization

  For memory distributed system