TOHOKU UNIVERSITY

GPPU GRADUATE PROGRAM ON PHYSICS FOR THE UNIVERSE

宇宙創成物理学国際共同大学院

APPLYING THE TRIAD NETWORK REPRESENTATION TO FOUR-DIMENSIONAL ATRG METHOD

Yuto Sugimoto , Shoichi Sasaki (Tohoku U.)

ABOUT TENSOR RENORMALIZATION GROUP

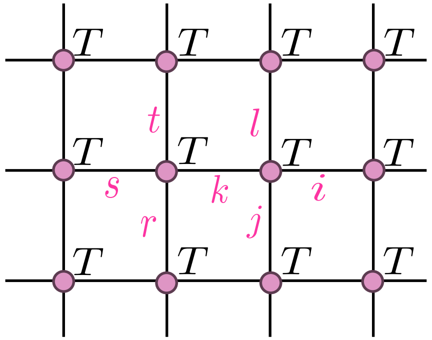
[M. Levin and C. P. Nave, (2007).]

Tensor Renormalization Group = Numerical Real space Renormalization Group

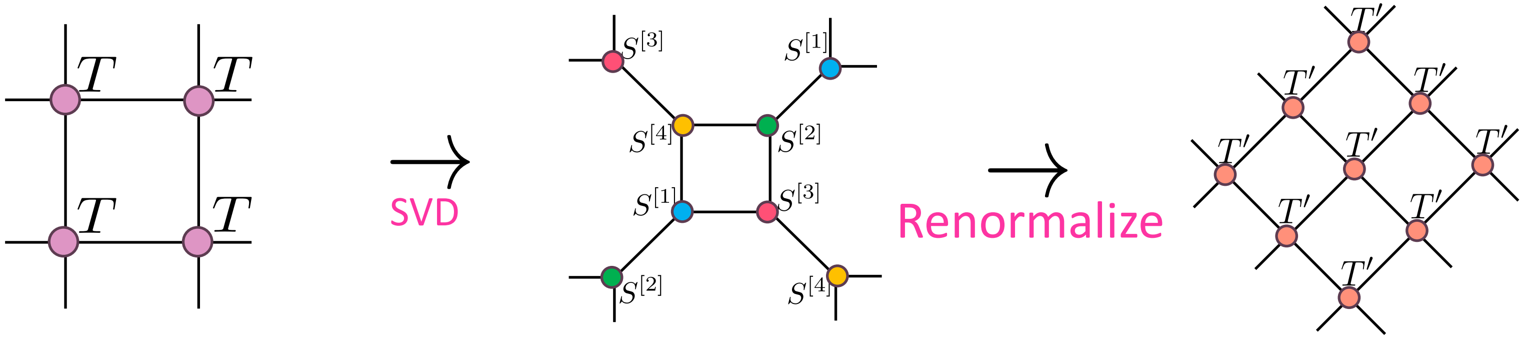
→A candidate for overcoming sign problems in LQCD

TRG is applicable for sign problem region, but has large cost at higher dimensions

$$Z = \sum_{i,j,k,l,m,\ldots}^{D} T_{ijkl} T_{mnoj} T_{krst} T_{opqr} \cdots = \mathrm{tTr}(\otimes_{x=1}^{N} T) =$$



Approximate by Singular Value Decomposition (SVD)



OTHER APPROACHES TO HIGHER DIMENSIONS

	HOTRG [Xie et al, (2012).]	ATRG [D. Adachi, T. Okubo, and S. Todo, (2020).]	Triad TRG [D. Kadoh and K. Nakayama, (2019).]	MDTRG Triad rep. [K. Nakayama, (2023).]
cost	$O(\chi^{4d-1})$	$O(\chi^{2d+1})$	$O(\chi^{d+3})$	$O(qr^3\chi^{d+3})$
Fundamental tensor	$O(\chi^{2d})$	$O(\chi^{d+1})$	$O(\chi^3)$	$O(\chi^{d+1})$
methods	• Exact Contraction	• Bond-swapping via RSVD • Exact Contraction	• Triad • Contraction via RSVD	• Decomposition of unit-cell tensor • Triad • Internal line oversampling • Contraction via RSVD

Our motivation is to search for a more efficient algorithm for four-dimensional theories.

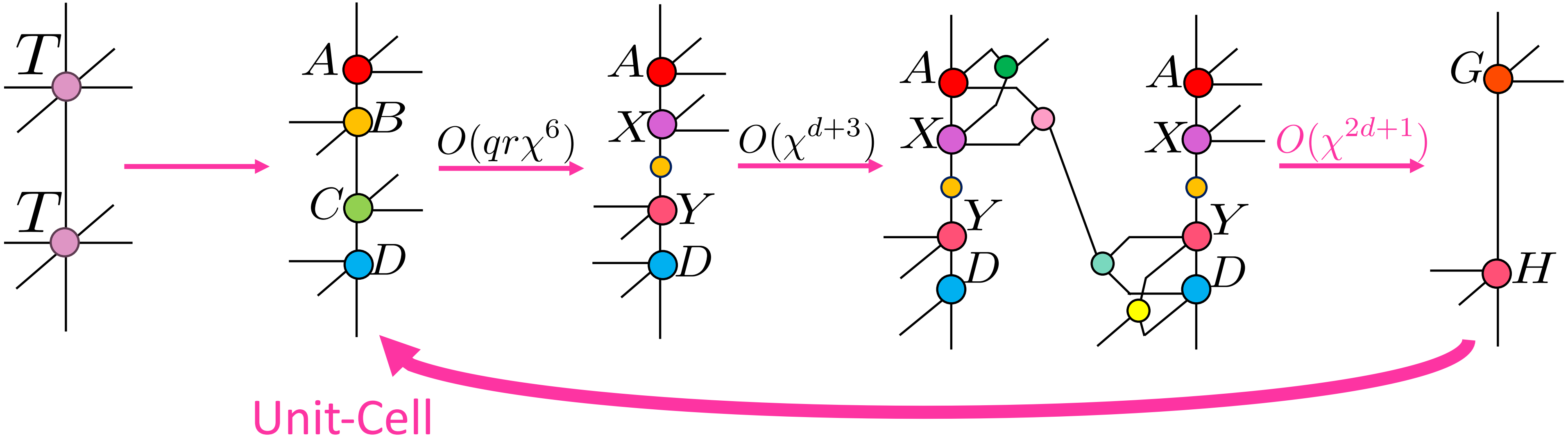
COST REDUCTION FOR HIGHER DIMENSIONS –ATRG

[D. Adachi, T. Okubo, and S. Todo, (2020).]

Two reason for the cost reduction

- ✓ The fundamental tensor has d+1 legs.
- ✓ By performing bond swapping, the number of isometries is reduced by 1/2.

The cost of ATRG is χ^{2d+1} ! (HOTRG was χ^{4d-1}) RSVD is used in the bond-swapping step



COST REDUCTION FOR HIGHER DIMENSIONS –MDTRG TRIAD REP.

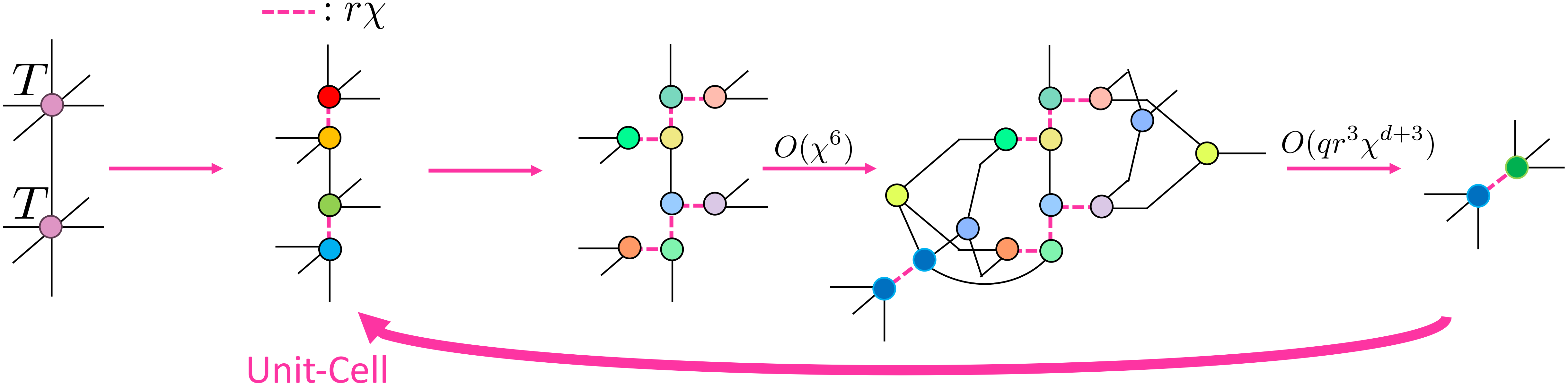
[K. Nakayama, (2023).]

Difference from ATRG is

- ✓ Using RSVD with QR iteration in contraction step (approximated SVD scheme)
- ✓ Oversampling of internal line $\chi \to r\chi$
- ✓ Decomposition of Unit-cell tensor

MDTRG is more accurate than Triad TRG , almost same accuracy of HOTRG

→ MDTRG-Triad rep. is improved version of Triad TRG, which cost is $O(qr^3\chi^{d+3})$



CONSIDERATION ON VARIOUS METHOD IN 4D

- In 4D systems, it is Trade-off between accuracy and computation cost

	HOTRG [Xie et al, (2012).]	ATRG [D. Adachi, T. Okubo, and S. Todo, (2020).]	MDTRG Triad rep. [K. Nakayama, (2023).]
cost	$O(\chi^{15})$ 😰	$O(\chi^{9})$ 🙂	$O(qr^{3}\chi^{7})$ 😁
Accuracy	🙂	😐	?
Problem	• Large cost, difficult to enlarge χ	• Large cost • The convergence of free energy is not as good in the 2D cases.	• Investigation is needed

PROPOSAL

We aim for faster algorithms!

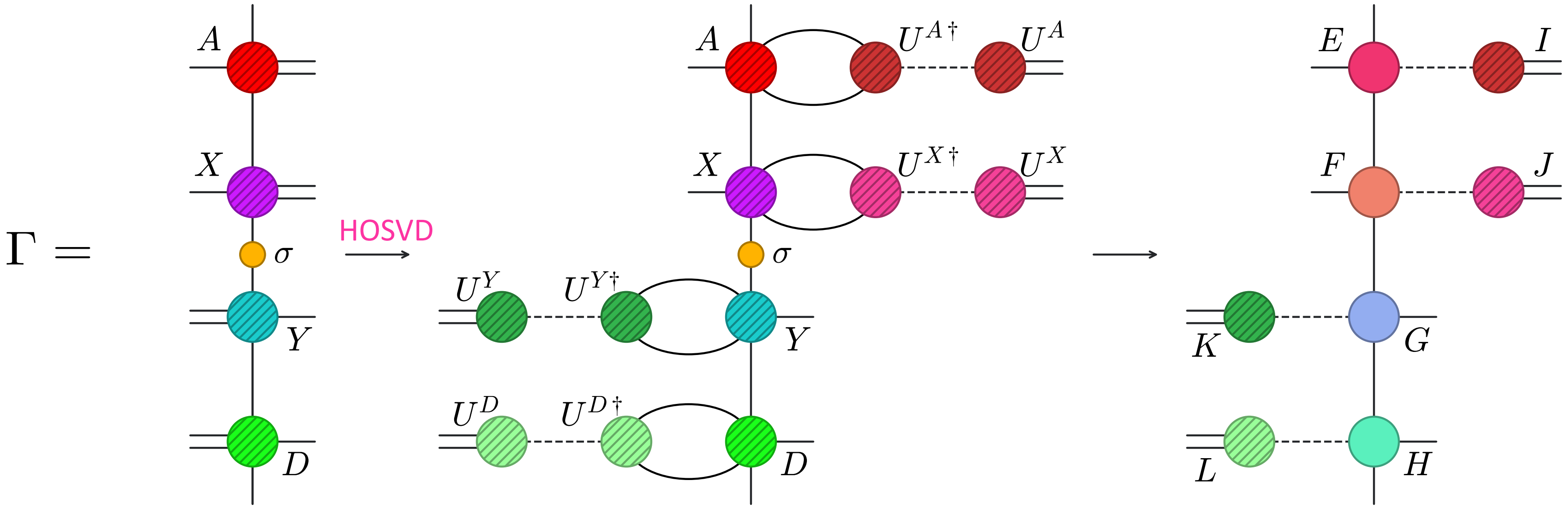
	ATRG [D. Adachi, T. Okubo, and S. Todo, (2020).]	Triad-MDTRG [K. Nakayama, (2023).]
Cost in 4D	$O(\chi^9)$	$O(qr^3\chi^7)$
methods	• Bond-swapping via RSVD • Exact Contraction • 3 isometry in the contraction step	• Decomp. of Unit-cell tensor • Triad • Internal line oversampling • Contraction via RSVD

Triad-ATRG ?

Research

TRIAD REPRESENTATION OF ATRG

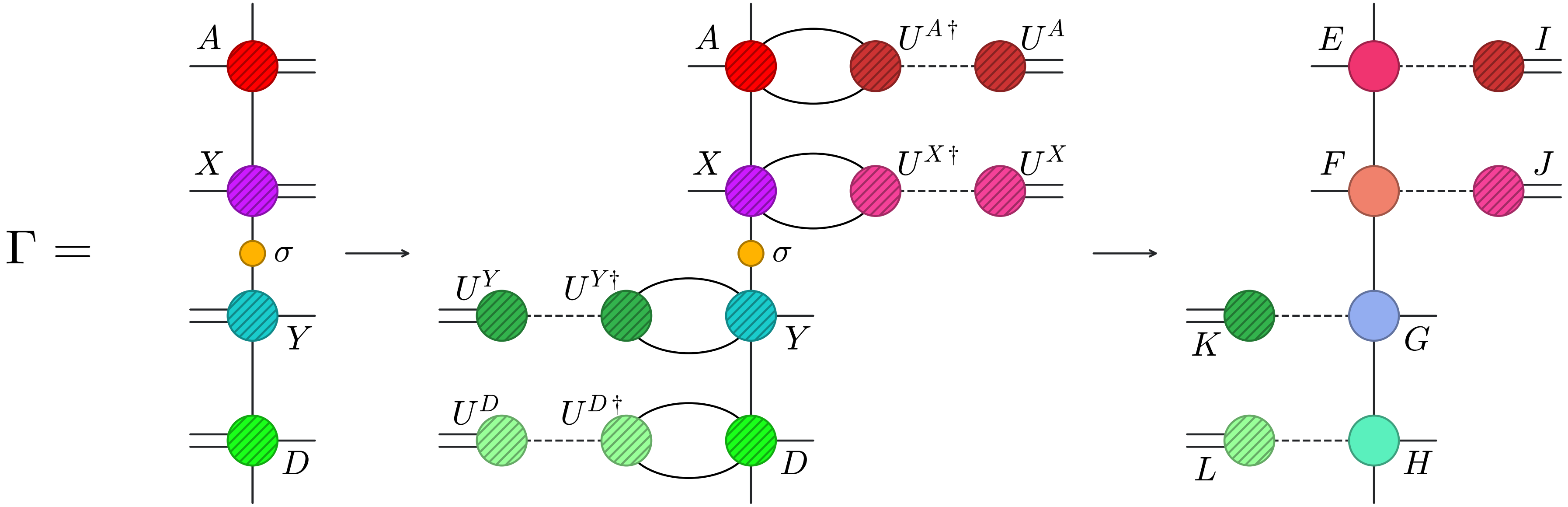
- We consider triad representation of ATRG
- Consider HOSVD of unit cell tensor $\Gamma = AX\sigma YD$ after the Bond swapping
- SVD of. $AX\sigma$ and σYD provides SVD of Γ thanks to canonical form [S. Akiyama, phd, 2022.]
- Make triad representation in the same manner as MDTRG
- Triad legs are oversampled $\chi \to r\chi$



TRIAD REPRESENTATION OF ATRG

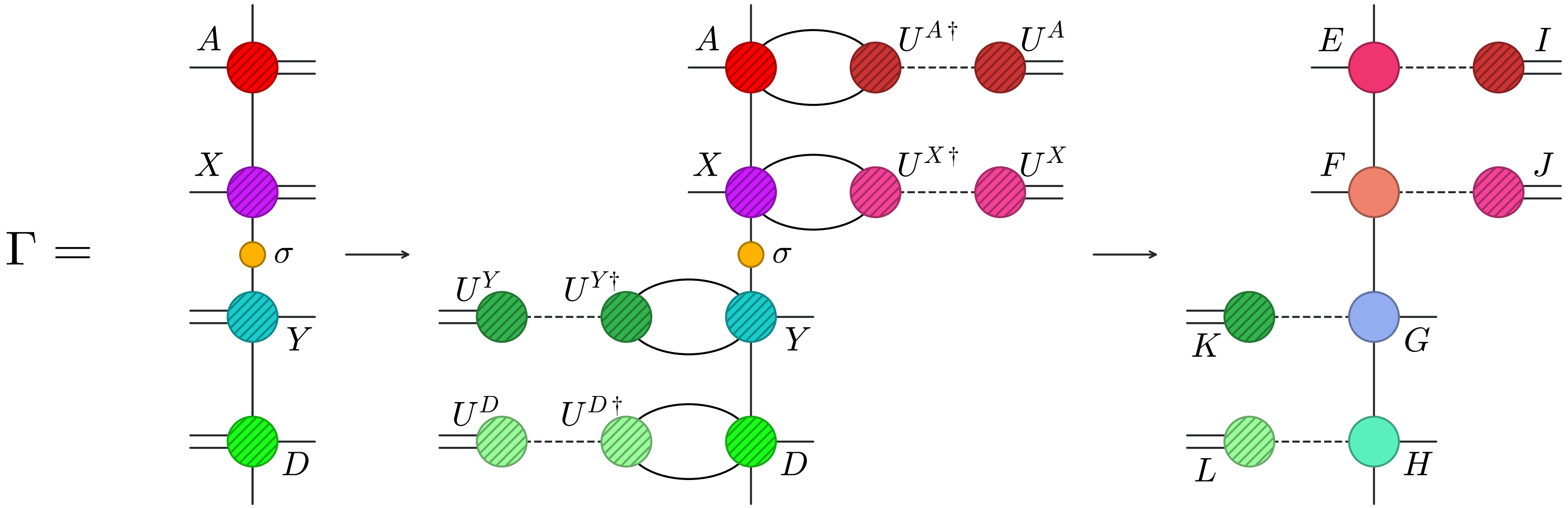
- However, in four dimensions, the order of computational cost does not change even if not all tensors are converted into triad form. Therefore, we use a form with as few decompositions as possible.
- We obtain 4 legs tensors $E, F, G, H \in \mathbb{C}^{\chi\times\chi\times\chi\times r\chi}$ and 3 legs tensors $I, J, K, L \in \mathbb{C}^{\chi\times\chi\times r\chi}$ (we call this form as triad rep.)

oversampled



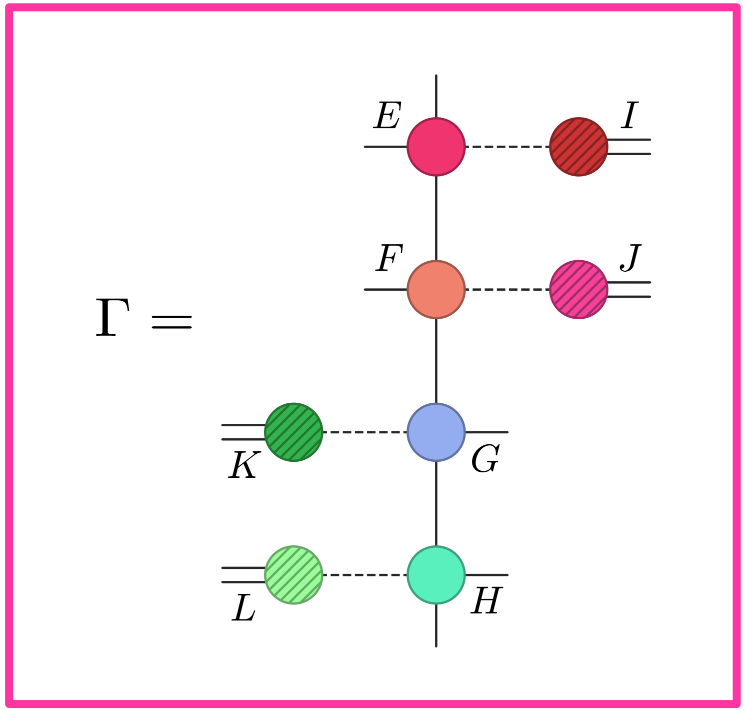
TRIAD REPRESENTATION OF ATRG

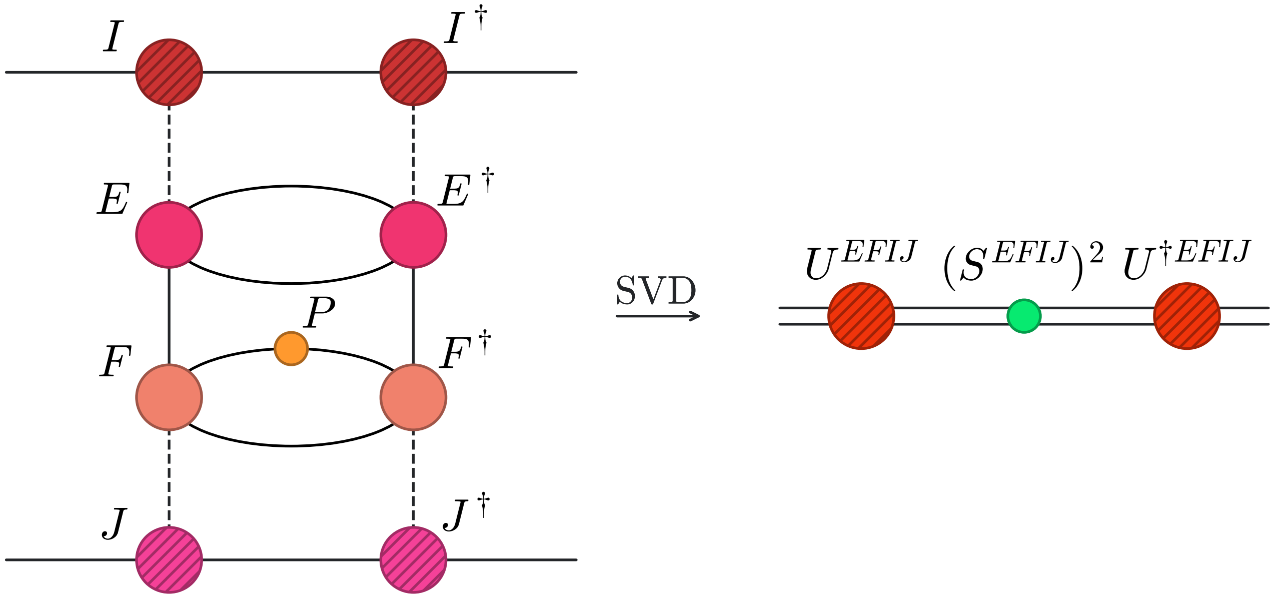
- Computational cost of this procedure is $O(\chi^7)$ (If we use RSVD, $O(qr\chi^6)$)
- All decomposition in this procedure are SVD of the Unit-Cell Tensor



MAKING SQUEEZERS

- We derive squeezers in the same manner of ATRG [S. Akiyama, phd, 2022.]
- since Γ is not canonical form anymore, we must decompose $\Gamma \simeq EFGHIJKL$
- We can calculate separately by introducing theGramm-matrix of. $EFIJ$ and $GHKL$





$$P = (GHKL)(GHKL)^\dagger$$

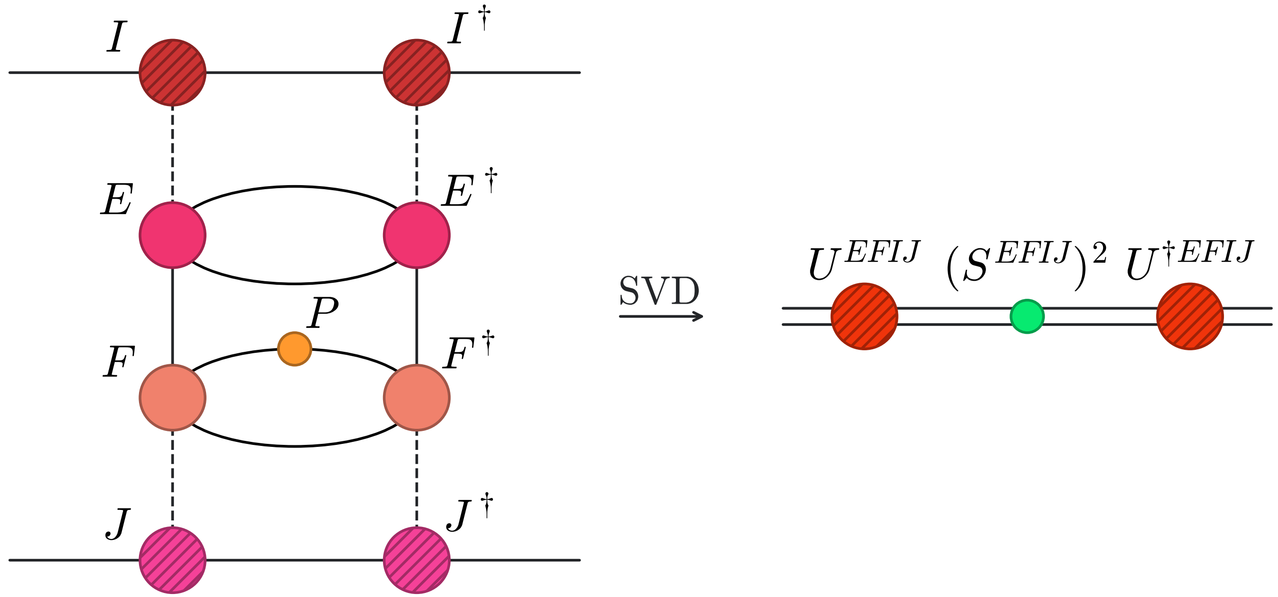
MAKING SQUEEZERS

- Computational cost of this procedure is $\min(O(\chi^7), O(r^2\chi^6))$
- All decomposition in this procedure are SVDs of $\Gamma(n)\Gamma(n+\hat{\mu})$ as in the improved ATRG

[S. Akiyama, phd, 2022.]

[S. Iino, S. Morita, and N. Kawashima, (2019).]

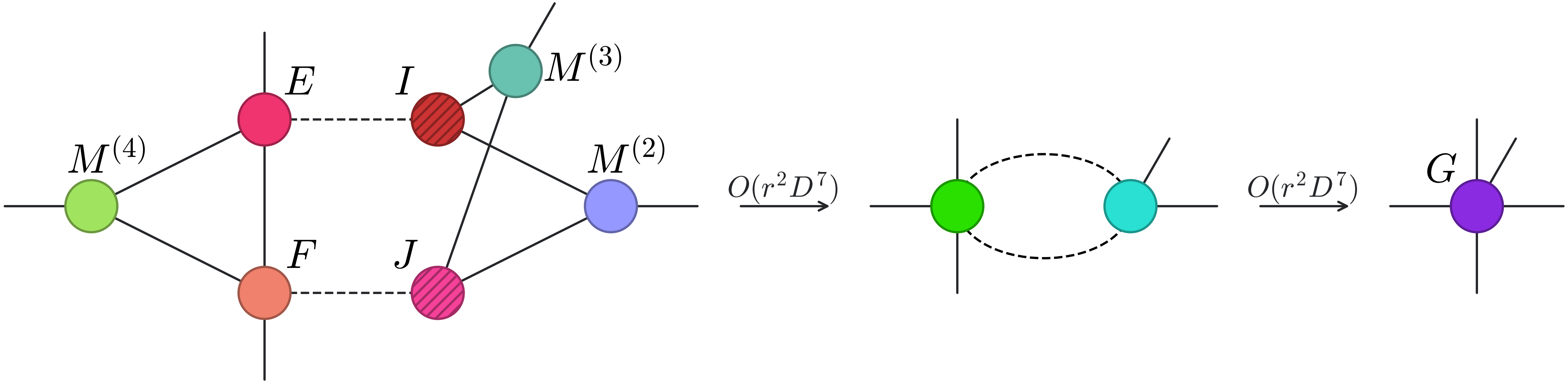
[D. Adachi, T. Okubo, and S. Todo, (2022).]



$$P = (GHKL)(GHKL)^\dagger$$

CONTRACTION STEP

- Thanks to the Triad form, Computational cost is reduced to $\underbrace{O(r^2\chi^7)}_{\text{Bottleneck}}$, smaller than ATRG ($O(\chi^9)$)
- We do not use RSVD since we already used it once in the bond-swapping step



SUMMARY OF COMPUTATIONAL COST

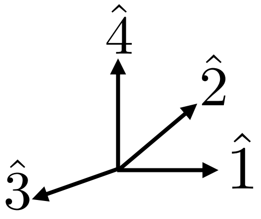
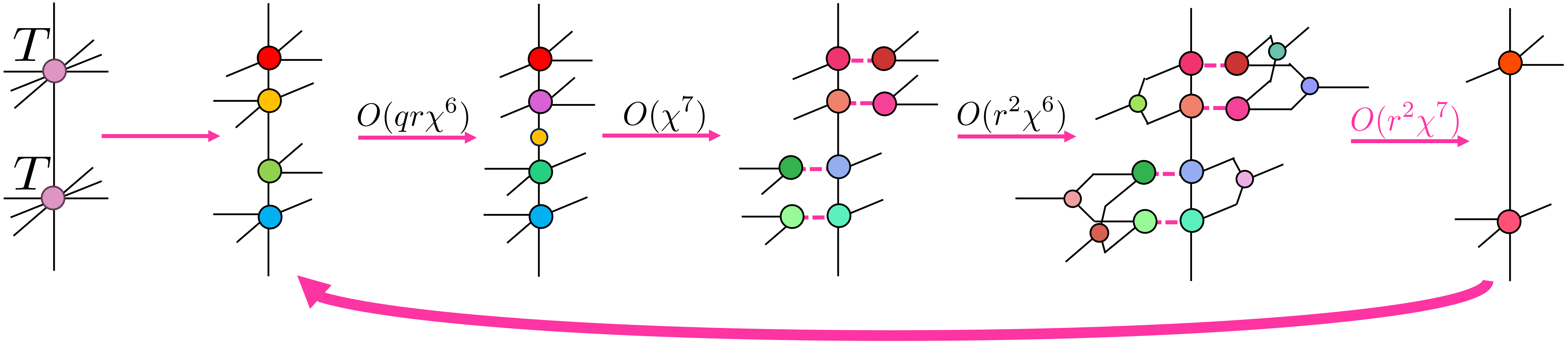


表1 Computational cost

Step	ATRG	Triad ATRG
Bond Swapping	$O(qr\chi^6)$	$O(qr\chi^6)$
Make Triad	None	$O(\chi^7)$
Squeezer	$O(\chi^7)$	$O(\min(\chi^7, r^2\chi^6))$
Contraction	$O(\chi^9)$	$O(r^2\chi^7)$

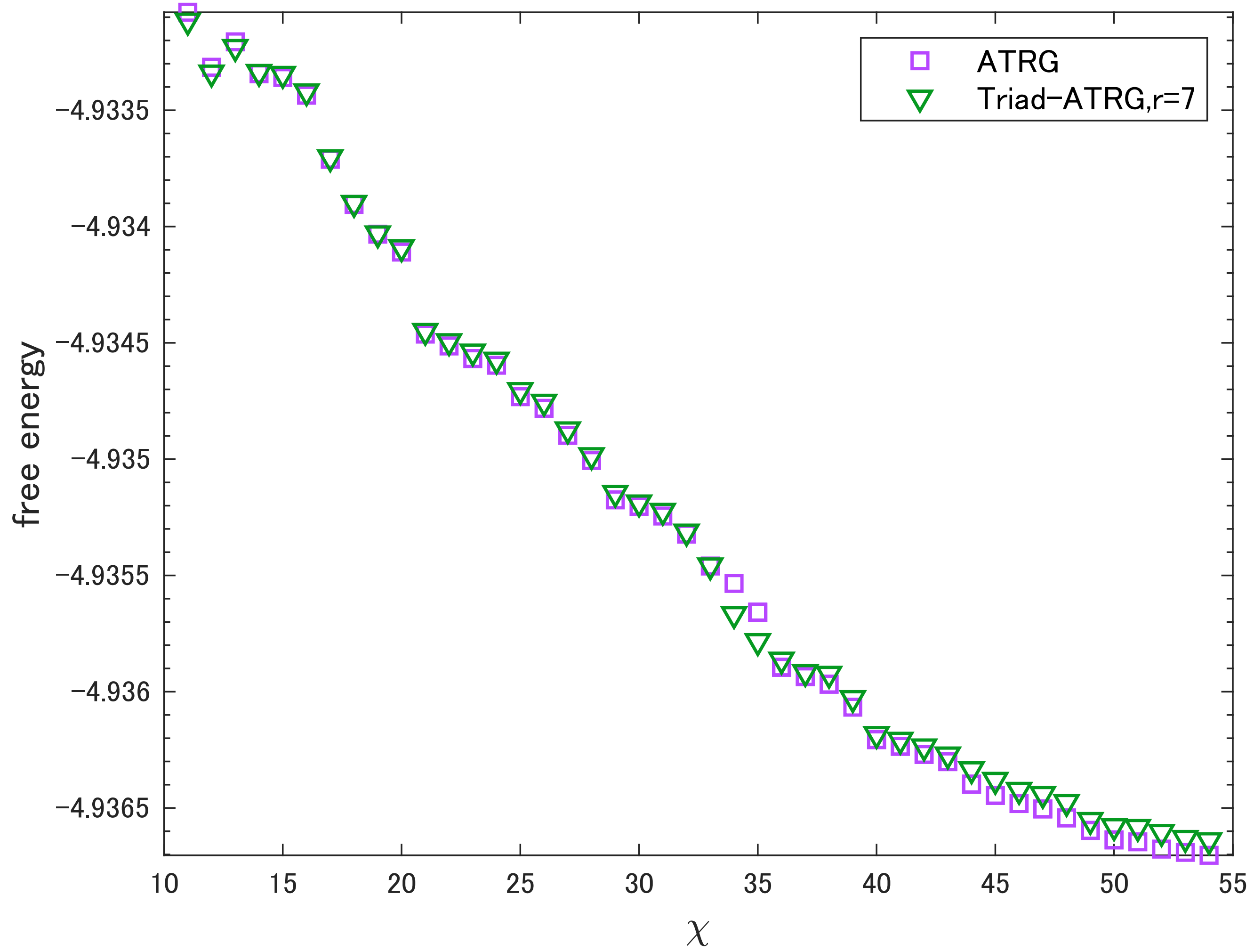
Bottleneck



Numerical results on 4D Ising model

FREE ENERGY

- We investigate the convergence of free energy at 4D Ising model in r=7, L=1024,T=6.65035
- The results are in high agreement with ATRG



HOTRG(D=13): $T_c = 6.650365(5)$

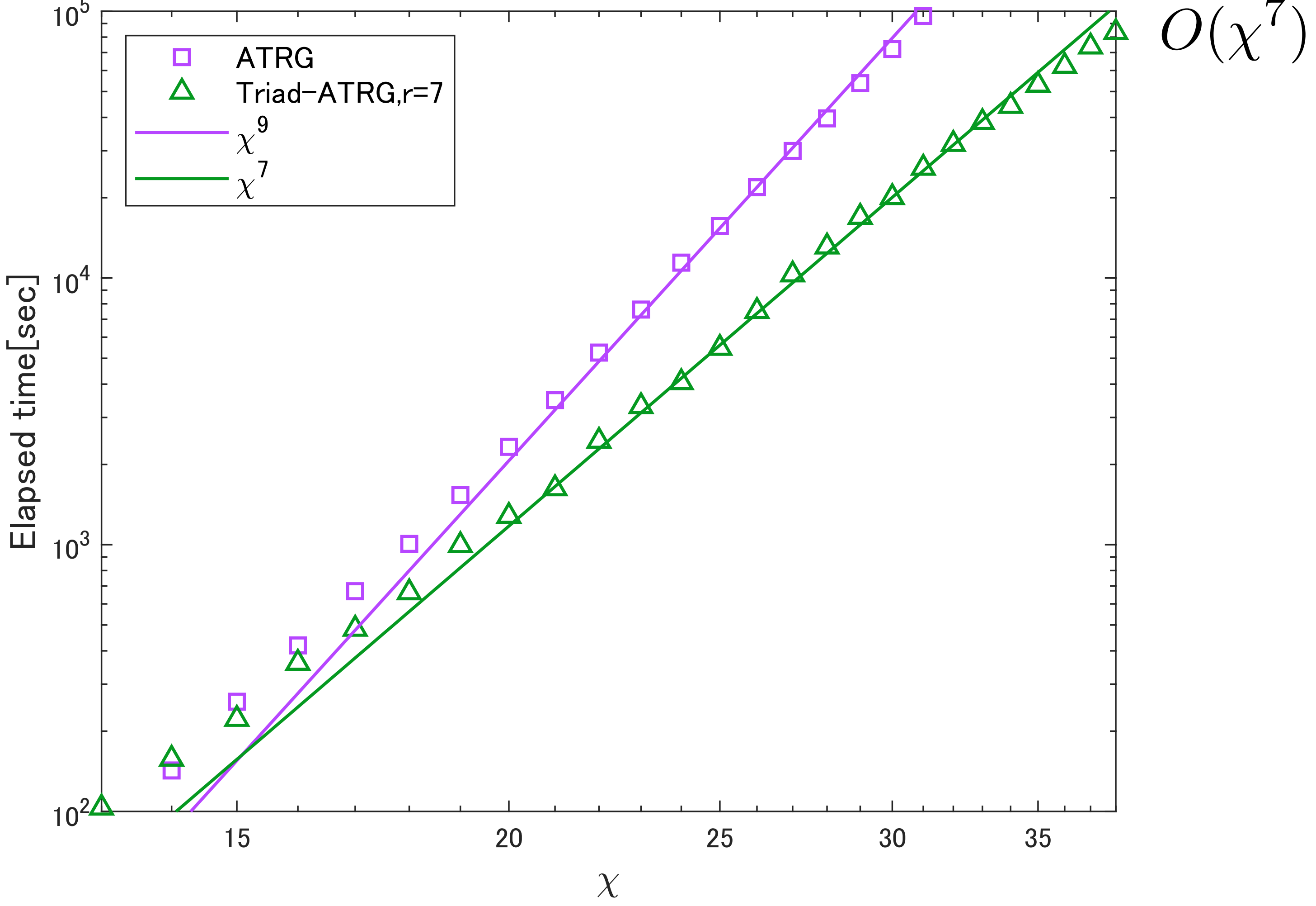
[S. Akiyama, Y. Kuramashi, T. Yamashita, and Y. Yoshimura, (2019).]

χ	ATRG	Triad-ATRG
46	-4.9364809	-4.9364227
47	-4.9365041	-4.9364402
48	-4.9365426	-4.9364745
49	-4.9365964	-4.9365523
50	-4.9366373	-4.9365787
51	-4.9366456	-4.9365820
52	-4.9366769	-4.9366039
53	-4.9366908	-4.9366297
54	-4.9367035	-4.9366392

Difference is only 0.0019%

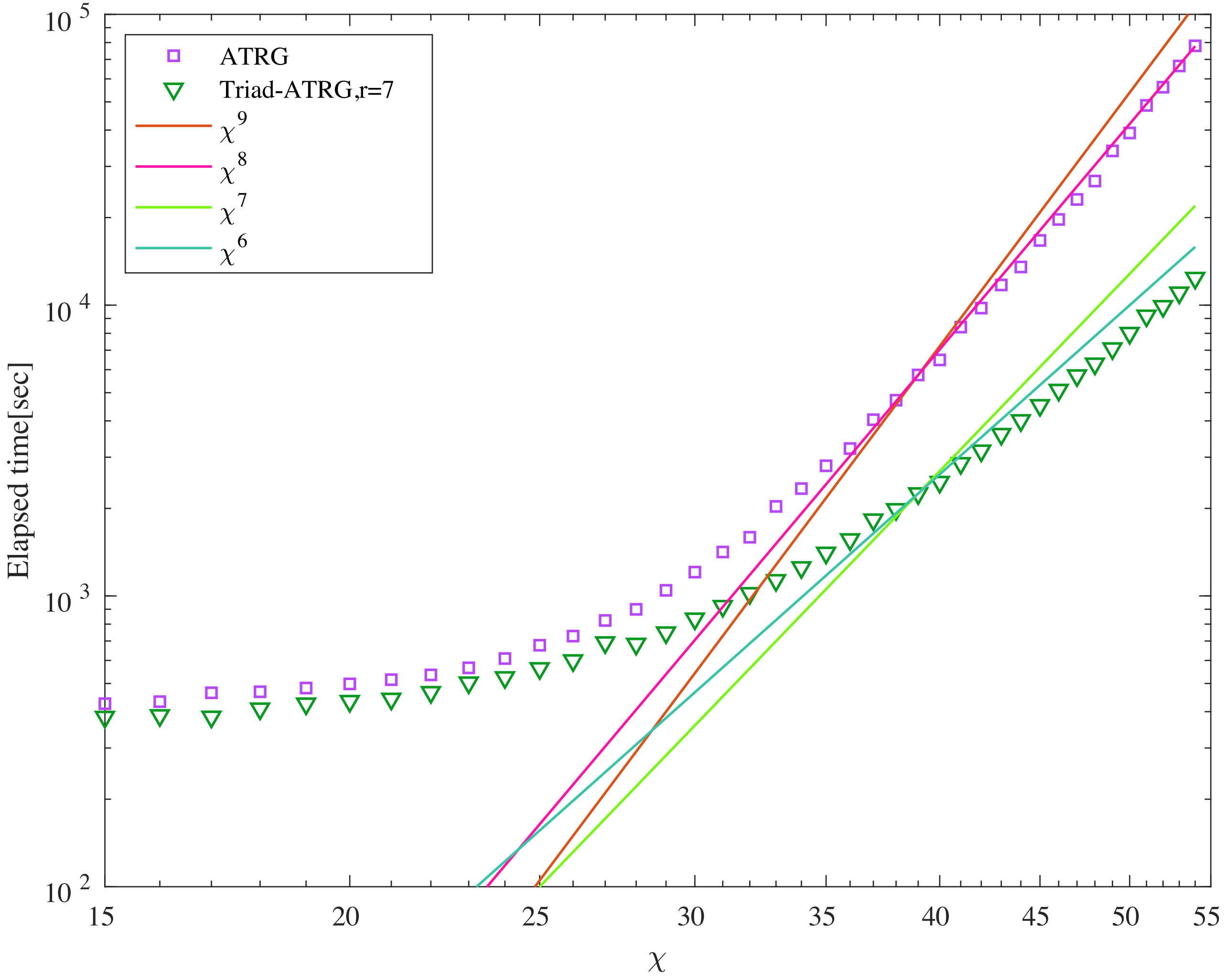
COMPUTATIONAL TIME ON A CPU

- We investigate the computational time in r=7 using a single CPU calculation
- Scaling of the computational time is $O(\chi^7)$



COMPUTATIONAL TIME ON GPUS

- We investigate the computational time in r=7 ,L=1024 by 2 GPU parallelized calculation
- Scaling of the computational time improved significantly



$\simeq O(\chi^8)$

$< O(\chi^6)$

Triad-ATRG could be powerful tool for GPU computations

We have used Tesla V100 16GB PCIE✕2

https://www.elsa-jp.co.jp/wp-content/uploads/2019/03/nvidia_tesla_v100_3qtr.png

PHASE TRANSITION POINT

To determine the transition point, we evaluate the following value at each coarse-graining step.

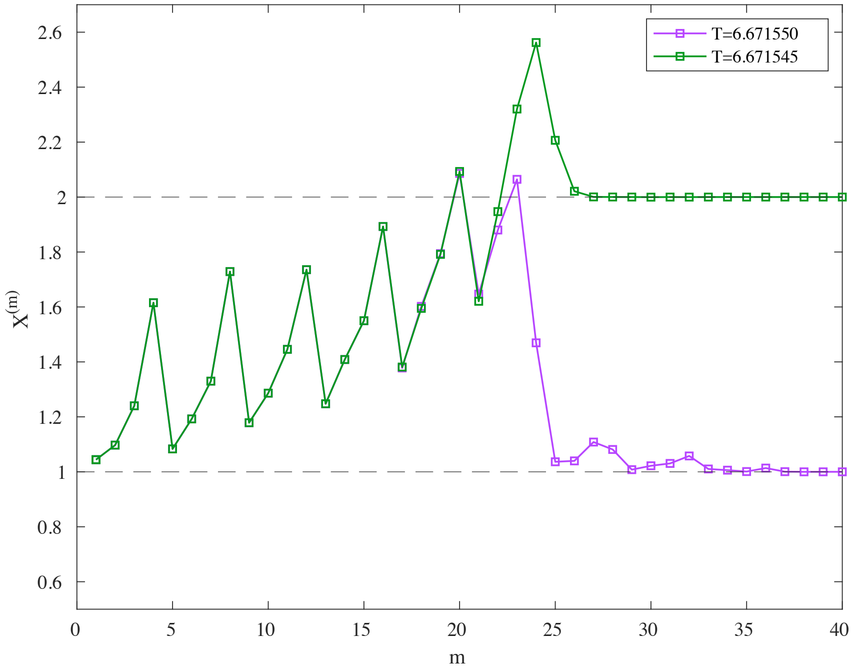
$$X^{(m)} = \frac{(\mathrm{Tr}A^{(m)})^2}{\mathrm{Tr}(A^{(m)})^2}, \text{ with } A_{kl}^{(m)} = \sum T^{(m)}_{i_1 i_2 i_3 k i_1 i_2 i_3 l}$$

HOTRG(D=13): $T_c = 6.650365(5)$

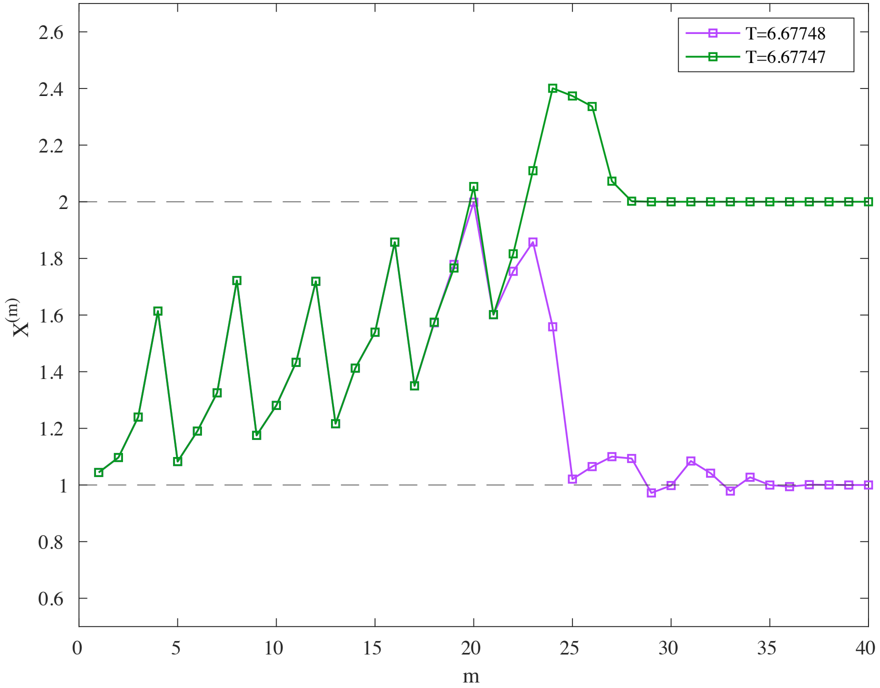
[S. Akiyama, Y. Kuramashi, T. Yamashita, and Y. Yoshimura, (2019).]

[Z.-C. Gu and X.-G. Wen, (2009).]

If X=2, the system is in an ordered phase, and if X=1, it is in a disordered phase

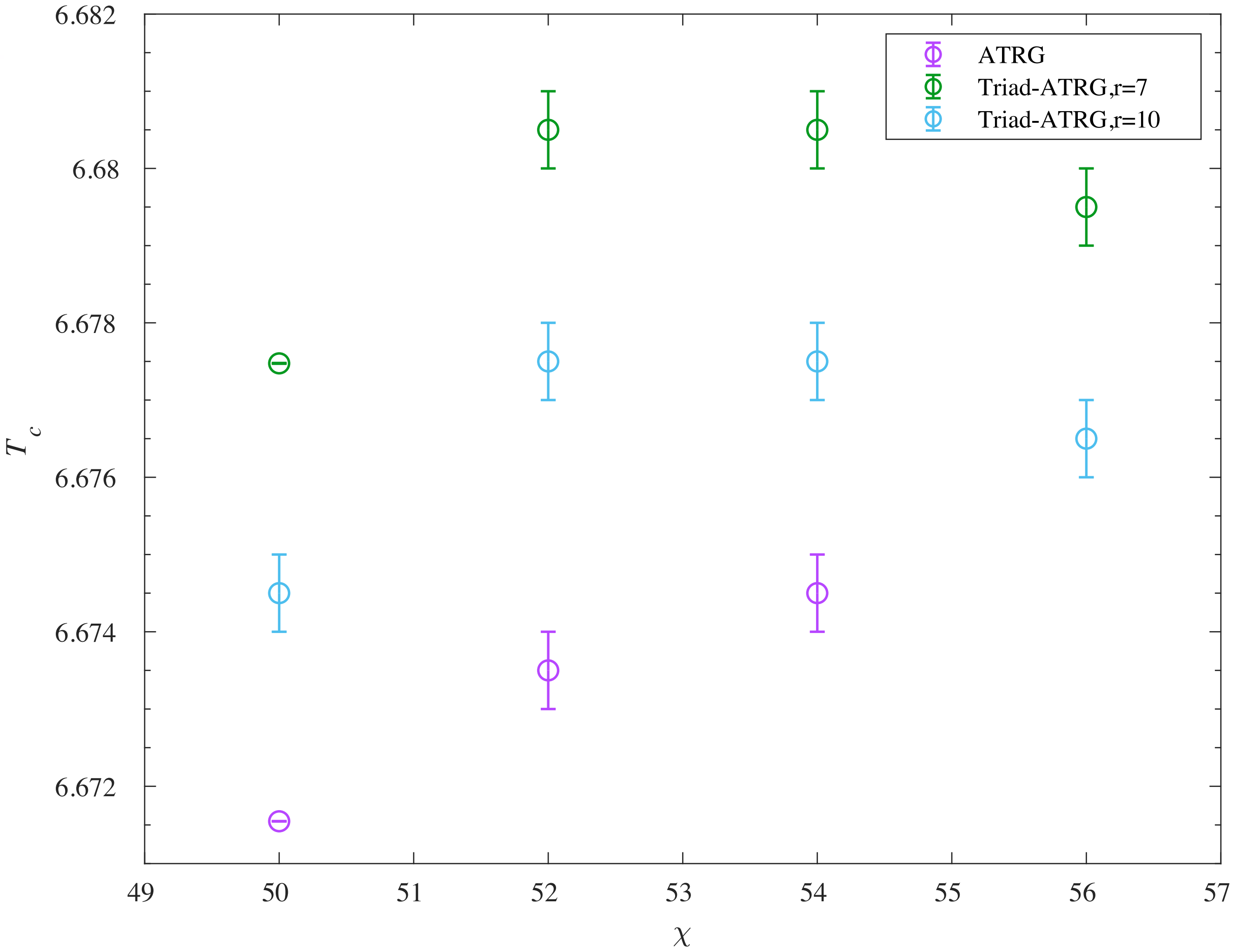


ATRG



Triad ATRG

PHASE TRANSITION POINT



HOTRG(χ=13): $T_c = 6.650365(5)$

[S. Akiyama, Y. Kuramashi, T. Yamashita, and Y. Yoshimura, (2019).]

Monte-Carlo: $T_c = 6.6803069(58)$

[P. H. Lundow and K. Markström, (2023).]

Difference from the ATRG results at χ=54 is ~0.1% for r=7, and ~0.04% for r=10

*results of ATRG has not converged well

Further investigation for larger χ is needed

SUMMARY AND FUTURE WORKS

- The results of Triad-ATRG are highly consistent with the ATRG results
- Triad-ATRG significantly improves the computational cost on CPU and GPUs

Triad ATRG would be a powerful tool for 4D systems

Future works

- Investigate the accuracy of internal energy
- Calculate in more large χ
- Apply to other 4D systems

END

ABOUT TENSOR NETWORK

The tensor network method was developed in condensed matter physics (Hamiltonian formalism)
For Quantum Field Theory, we use Lagrangian approach

	Hamiltonian	Lagrangian
Target	Many body system	Path integral
Method	Variational	Renormalization
Physics	Ground state Real time evolution	Partial function Green function

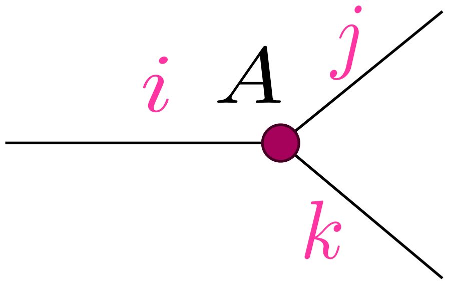
We use TRG for higher dimensions

ABOUT SUMMATION RULES

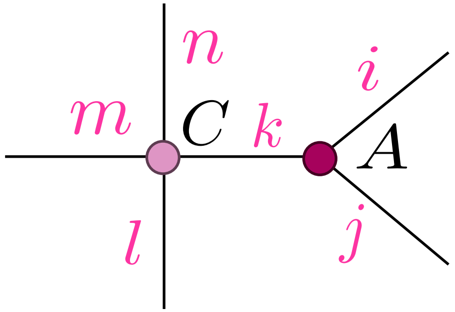
Tensor is multidimensional array!

We use convention of tensor network diagrams below.

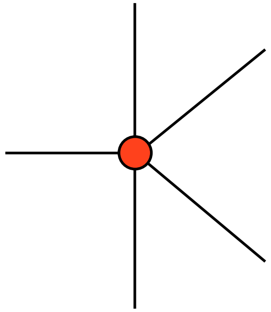
$$A_{ijk} =$$



$$\sum_{k}^{\chi} C_{klmn} A_{kij} =$$

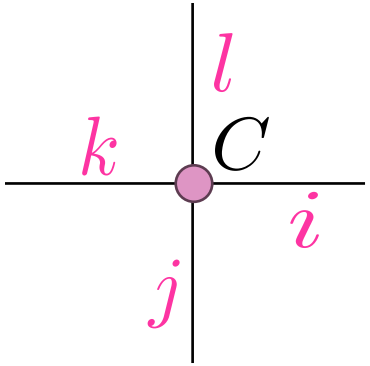


$$=$$



Summation rule

$$C_{ijkl} =$$



Tensor Network = Diagram representation of tensor contraction

COARSE GRAINING AND SINGULAR VALUE DECOMPOSITION

Theorem Let $m, n \in \mathbb{N}$ Any comprex matrix with rank r, $A \in \mathbb{C}^{m\times n}$ is decomposed by using singular value $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r \geq 0$,

$$A = U\Sigma V^\dagger \tag{1}$$

with $U \in \mathbb{C}^{m\times m},\ V \in \mathbb{C}^{n\times n}$is unitaly matrix, and

$$\Sigma = \begin{pmatrix} \mathrm{diag}(\sigma_1, \ldots, \sigma_r) & O_{r,m-r} \\ O_{m-r,r} & O_{m-r,n-r} \end{pmatrix} \tag{2}$$

$$AA^\dagger = U\Sigma^2 U^\dagger, \quad A^\dagger A = V\Sigma^2 V^\dagger$$

Singular values have important information of the matrix. (next page)

BEST LOW RANK APPROXIMATION

Theorem The comprex matrix $X \in \mathbb{C}^{m\times n}$ which satisfies $||A - X||_F = \min_{\text{rank}X'=k} ||A - X'||_F$with rank $k < r$ is

$$||A||_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2}$$

$$X = \sum_{m=1}^{k} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^\dagger \tag{1}$$

$$= \begin{pmatrix}\boldsymbol{u}_1 & \cdots & \boldsymbol{u}_k\end{pmatrix} \begin{pmatrix}\sigma_1 & & \\ & \ddots & \\ & & \sigma_k\end{pmatrix} \begin{pmatrix}\boldsymbol{v}_1^\dagger \\ \vdots \\ \boldsymbol{v}_k^\dagger\end{pmatrix} \tag{2}$$

$\boldsymbol{u}$ and $\boldsymbol{v}$ is columns of U, V

Truncated-SVD,(RSVD,Arnordi,…)

SVD is the best approximation for matrix

RANDOMIZED SVD

[N. Halko, P.-G. Martinsson, and J. A. Tropp, (2010).]

Let $A \in \mathbb{C}^{m\times n}$ be a target matrix, and let $\Omega \in \mathbb{C}^{n\times l}$ be a random matrix with $l = \chi + p$ with a target rank χ.
We compute $Q = \mathrm{qr}(A\Omega)$. Then we iterate the following procedure for q times:

$$\Omega = \mathrm{qr}(A^\dagger Q)$$
$$Q = \mathrm{qr}(A\Omega)$$

We obtain $A \simeq QQ^\dagger A$. Finally, by computing the SVD

$$Q^\dagger A \simeq \overline{U}\Sigma V^\dagger$$

we obtain the truncated SVD of $A \simeq Q\overline{U}\Sigma V^\dagger$ with $\Sigma \in \mathbb{C}^{\chi\times\chi}$.
Total cost is $O(qmn\chi)$

RSVD is powerful tool in the TRG calculations

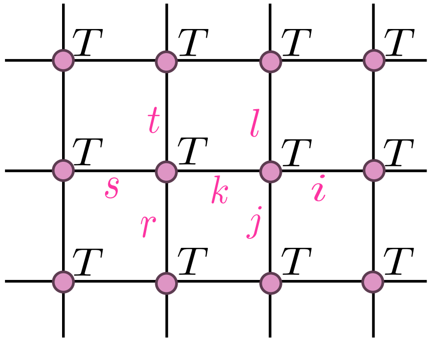
ABOUT TENSOR RENORMALIZATION GROUP

[M. Levin and C. P. Nave, (2007).]

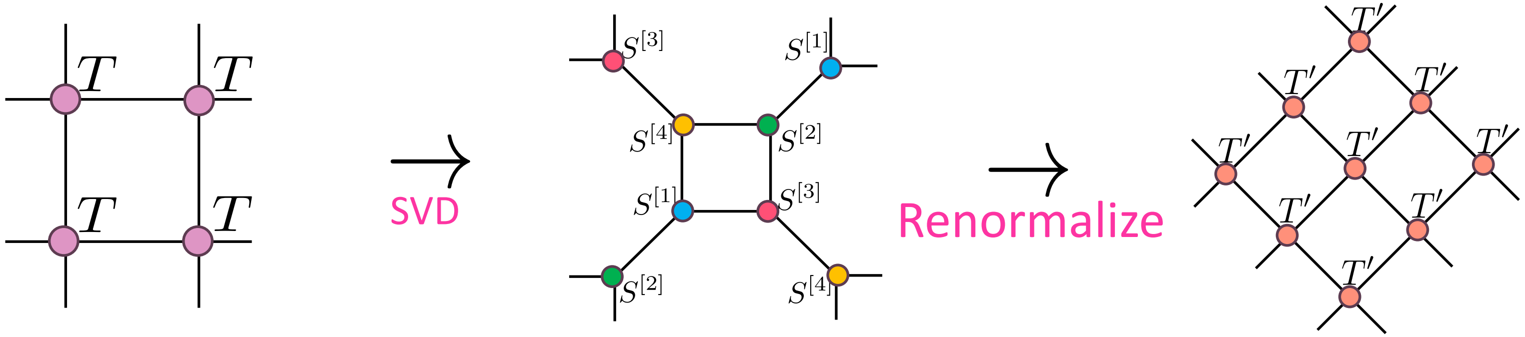
Tensor Renormalization Group = Numerical Real space Renormalization Group

Partial function can be expressed in Tensor network form if interaction is

$$Z = \sum_{i,j,k,l,m,\dots}^{D} T_{ijkl}T_{mnoj}T_{krst}T_{opqr}\cdots = \mathrm{tTr}(\otimes_{x=1}^{N} T) =$$



Approximate Z by Singular Value Decomposition (SVD)



SVD

Renormalize

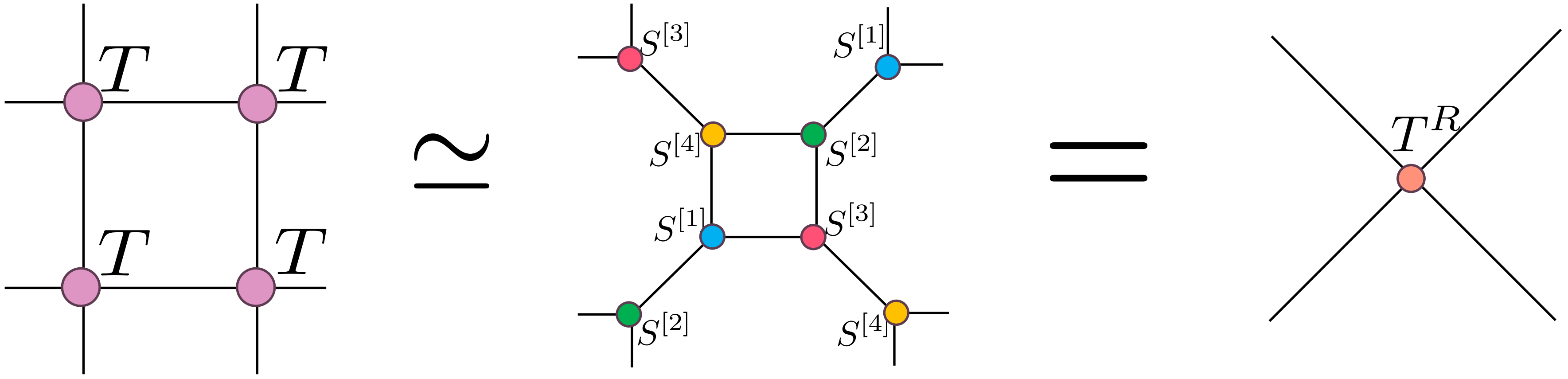
Tensor Renormalization Group (TRG) method has no sign problems !

TRG is applicable for sign problem region, but has large cost at higher dimensions

TENSOR RENORMALIZATION GROUP

[M. Levin and C. P. Nave, (2007).]

Applying SVD, we decompose subnetwork, and contract inner bonds $T \in \mathbb{C}^{\chi\times\chi\times\chi\times\chi}$



Rotating the system, we get new network, but the number of tensors became 1/4

→ Renormalization , total computational cost is $\chi^6 \log V$

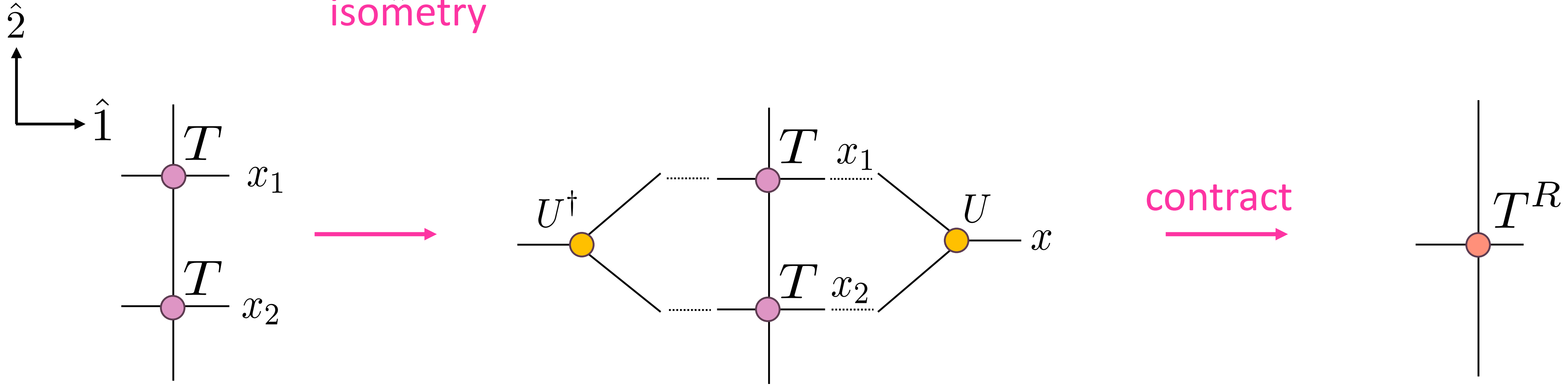
HOTRG

[Z. Y. Xie, J. Chen, M. P. Qin, J. W. Zhu, L. P. Yang, and T. Xiang, (2012).]

For higher dimensions, HOTRG renormalize each direction separately.

We multiply truncated unitary matrix $U_{x_1 x_2 x} \in \mathbb{C}^{\chi\times\chi\times\chi}$ from each side. $UU^\dagger \simeq I$

||
isometry



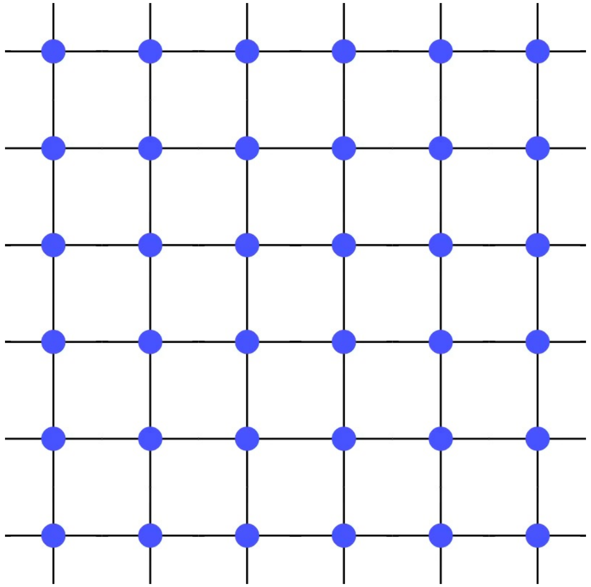
$$T^R_{i_1(n^*)i_2(n^*)j_1(n^*)j_2(n^*)} = U^*_{j_1(n+\hat{1})j_1(n)j_1(n^*)} T_{i_1(n+\hat{1})i_2(n+\hat{1})j_1(n+\hat{1})j_2(n+\hat{1})} T_{i_1(n)i_2(n)j_1(n)j_2(n)} U_{i_1(n+\hat{1})i_1(n)i_1(n^*)}$$

U is obtained by SVD of subnetwork TT ; this is optimal for y direction.

HOTRG PROCEDURE

Renormalize each direction anisotropically

Applicable to 3d,4d systems



https://smorita.github.io/TN_animation/

$$Z = \sum_{i,j,k,l,m,\ldots}^{\chi} \underbrace{T_{ijkl}T_{mnoj}}_{\text{Renormalize}} \underbrace{T_{krst}T_{opqr}}_{\text{Renormalize}} \cdots$$

$$\simeq \sum_{i,j,k,l,m,\ldots}^{\chi} \underbrace{T^{(1)}_{ijkl} \cdot T^{(1)}_{mnoj}}_{\text{Renormalize}} \cdots$$

$$\simeq \sum_{i,j,k,l,m,\ldots}^{\chi} T^{(2)}_{ijkl} \cdots$$

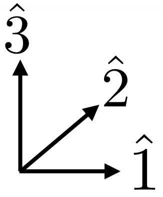
- $\mathrm{computational\,cost} = \chi^{\alpha} \log \mathrm{V} \rightarrow$ Easy to take thermodynamic limit ($\mathrm{MC} \rightarrow \mathrm{V}^{\beta}$)
- No Sign Problem
- Easy to treat Fermion

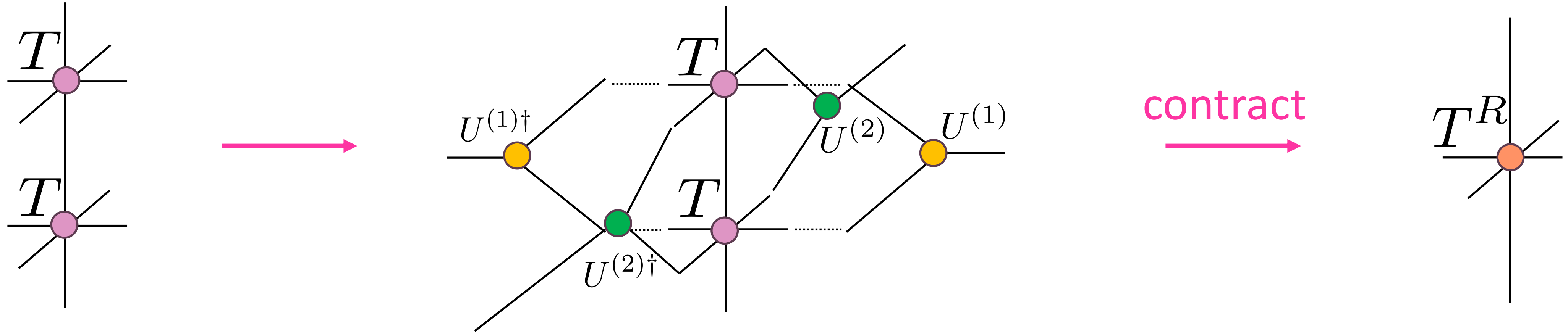
HOTRG

[Z. Y. Xie, J. Chen, M. P. Qin, J. W. Zhu, L. P. Yang, and T. Xiang, (2012).]

For 3d or higher dimensions, this procedure can be also done by separatory

We multiply truncated unitary matrices $U^{(1)}_{x_1 x_2 x},\quad U^{(2)}_{y_1 y_2 y}$ from x and y axis.





$$T^R = U^{*(2)} U^{*(1)} T T U^{(1)} U^{(2)}$$

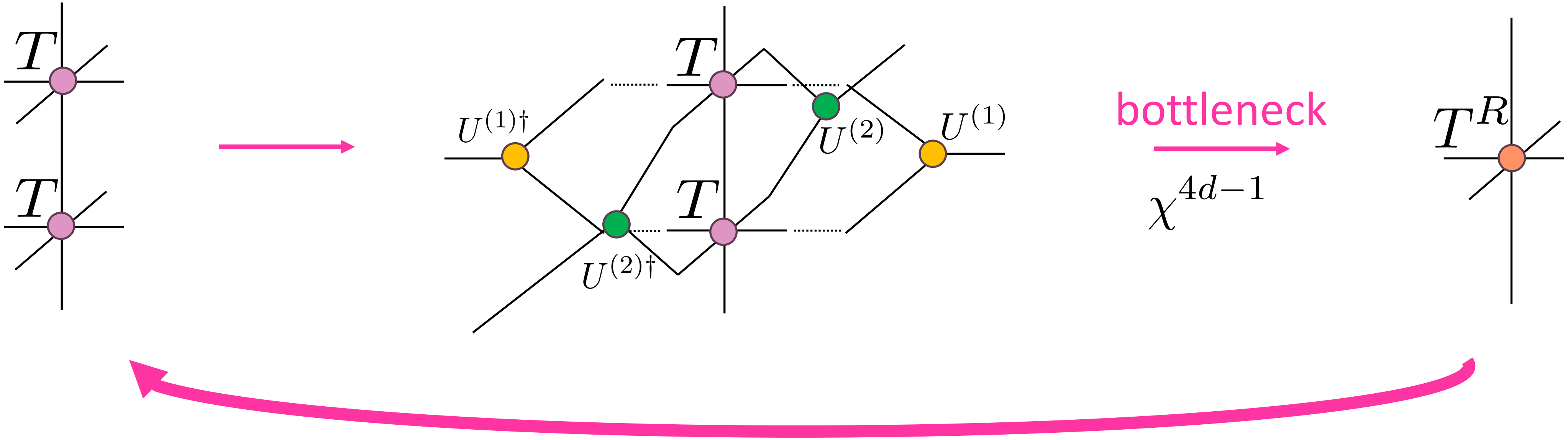
U is obtained by Eigen Value Decomposition of subnetwork TT

HOTRG

[Z. Y. Xie, J. Chen, M. P. Qin, J. W. Zhu, L. P. Yang, and T. Xiang, (2012).]

HOTRG can be applied to any dimensional systems, but it has large computational cost.

HOTRG has χ^{4d-1} for contraction step, it is too large for 4d systems like QCD

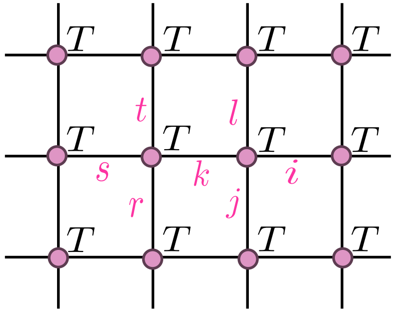


CONSTRACT TENSOR REPRESENTATION

Partial function can be expressed in Tensor network form if interaction is local

Nontrivial??

$$Z = \sum_{i,j,k,l,m,\dots} T_{ijkl} T_{mnoj} T_{krst} T_{opqr} \cdots =$$



We can constract Tensor Network Representations of the spin system as follows.

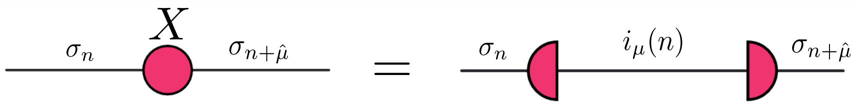
Consider following Hamiltonian

$$\text{Ising model} \quad W[\sigma_n, \sigma_{n+\hat{\mu}}] = -J\sigma_n \sigma_{n+\hat{\mu}}, \quad K[\sigma_n] = h\sigma_n$$

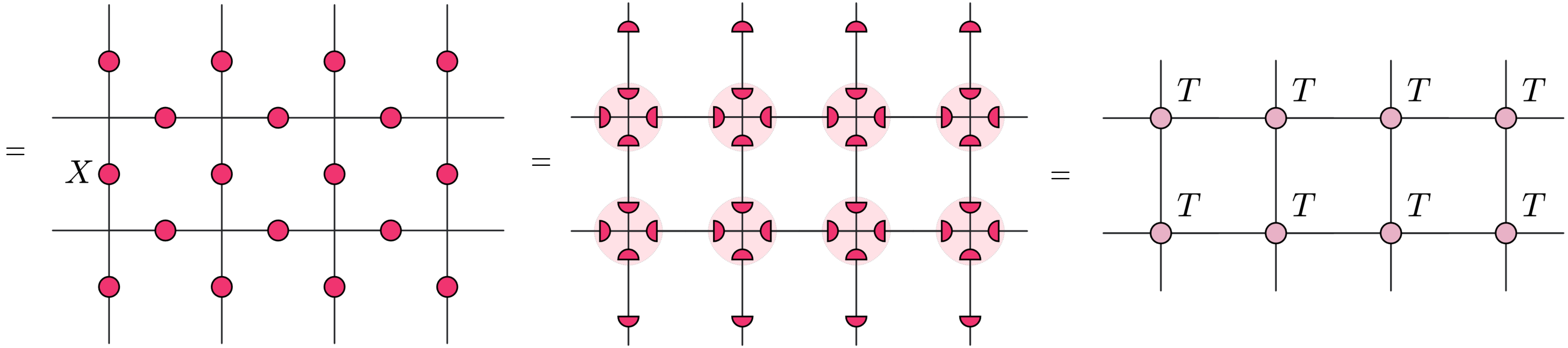
$$Z = \sum_{\sigma_n} \left[\prod_n \exp\left(\sum_\mu \underbrace{W[\sigma_n, \sigma_{n+\hat{\mu}}]}_{\text{hopping term}} + \underbrace{K[\sigma_n]}_{\text{on-site term}} \right) \right]$$

CONSTRACT TENSOR REPRESENTATION

Let $X_{\sigma_n\sigma_{n+\hat{\mu}}} = \exp\left(W[\sigma_n,\sigma_{n+\hat{\mu}}]\right) \overset{\text{SVD}}{=} \sum_{i_\mu(n)} U_{\sigma_n} S_{i_\mu(n)} V^*_{\sigma_{n+\hat{\mu}} i_\mu(n)}$



$$Z = \sum_{\sigma}\left[\prod_{n}\exp\left(\sum_{\mu}\underbrace{W[\sigma_n,\sigma_{n+\hat{\mu}}]}_{\text{hopping term}} + \underbrace{K[\sigma_n]}_{\text{on-site term}}\right)\right] = \sum_{\sigma}\left[\prod_{n}\prod_{\mu}X_{\sigma_n\sigma_{n+\hat{\mu}}}\right]\exp(K[\sigma_n])$$



$$T_{i_1(n)\ldots i_d(n) i_1(n-\hat{1})\ldots i_d(n-\hat{d})} = \sum_{\sigma_n} e^{K[\sigma_n]} \prod_{\mu}^{d} U_{\sigma_n i_\mu(n)} V^*_{\sigma_n i_\mu(n-\hat{\mu})} \sqrt{S_{i_\mu(n)}}\sqrt{S_{i_\mu(n-\hat{\mu})}}$$

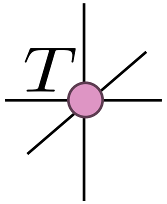
COST REDUCTION FOR HIGHER DIMENSIONS –ATRG

[D. Adachi, T. Okubo, and S. Todo, (2020).]

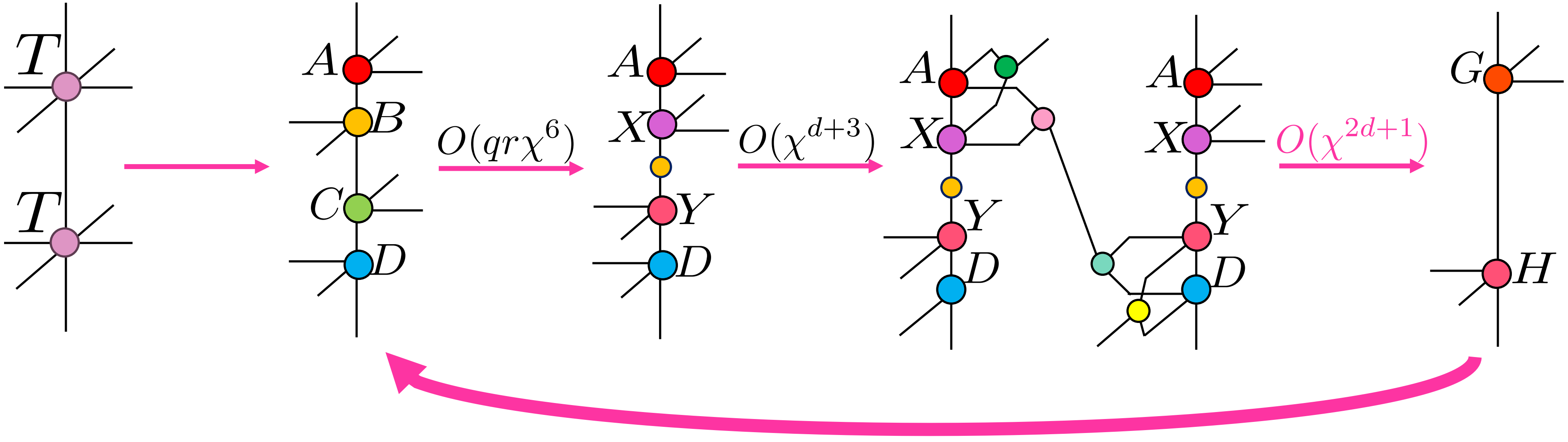
In d-dimensional system, The tensor T has 2d legs, which results in high memory cost!

One way to reduce cost is the ATRG method. -Adachi et al (2020)

1. Decompose a fundamental tensor to d+1 legs. $TT \rightarrow ABCD$
2. Decompose $BC \rightarrow X\sigma Y$ with the bond swapping
3. Using isometry (called squeezer), make GH and next $ABCD$



6 legs



COST REDUCTION FOR HIGHER DIMENSIONS –MDTRG

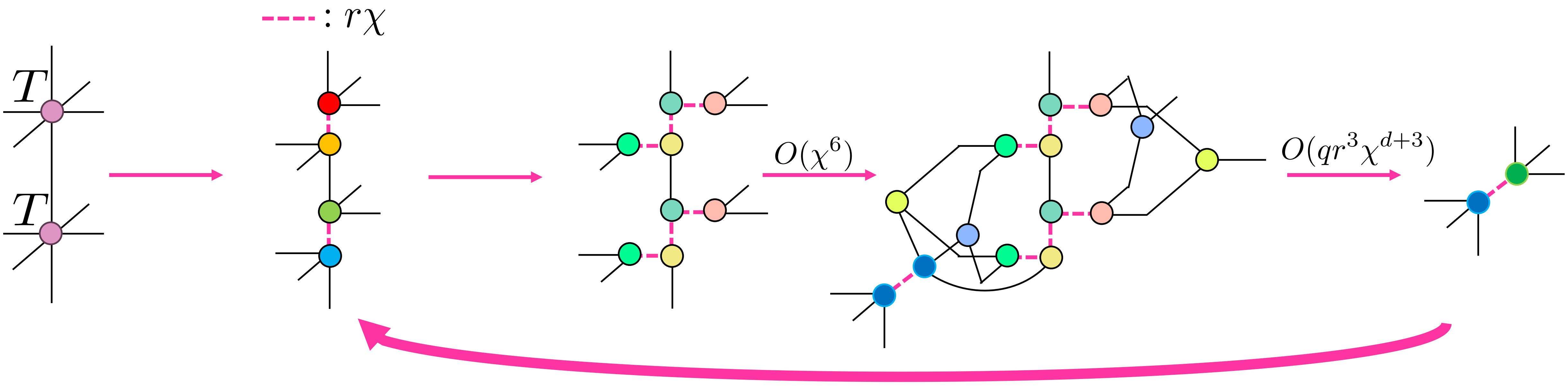
[K. Nakayama, (2023).]

MDTRG-Triad rep. is improved version of Triad TRG, which cost is $O(qr^3\chi^{d+3})$

Difference from Triad TRG is

- ✓ Oversampling of internal line $\chi \to r\chi$
- ✓ Using RSVD with QR iteration in contraction step (approximated SVD scheme)
- ✓ Decomposition of Unit-cell tensor

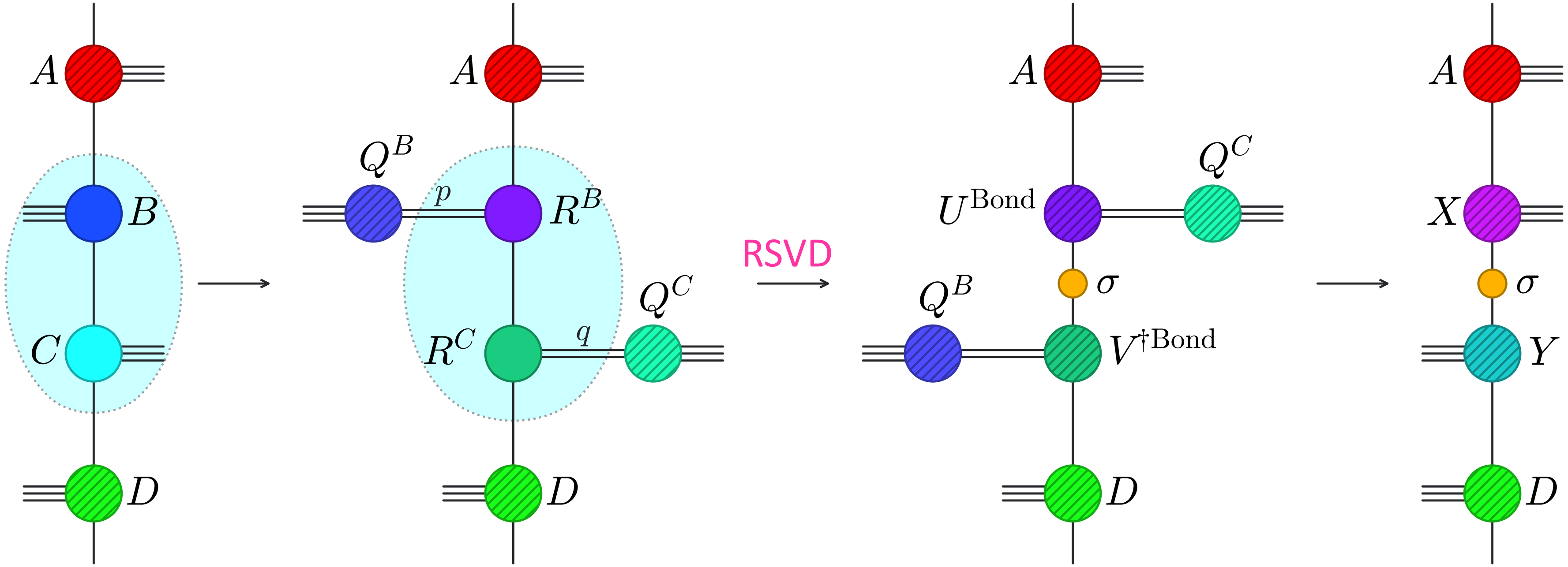
⟶ MDTRG is more accurate than Triad TRG , same accuracy of HOTRG



RSVD IN 4D SYSTEM

- The numerical instability caused by RSVD is serious in four dimensions [H. Oba, (2019).]
- In ATRG, $q = O(D)$ is sufficient in Bond swapping step, total cost is $O(\chi^9)$
- If we take $q = O(D)$ in MDTRG, the cost is $O(r^3\chi^8)$, difficult to enlarge χ and r simultaneously

→ Is there any efficient triad algorithms for 4D system ?



COST REDUCTION FOR HIGHER DIMENSIONS –TRIAD TRG

[D. Kadoh and K. Nakayama, (2019).]

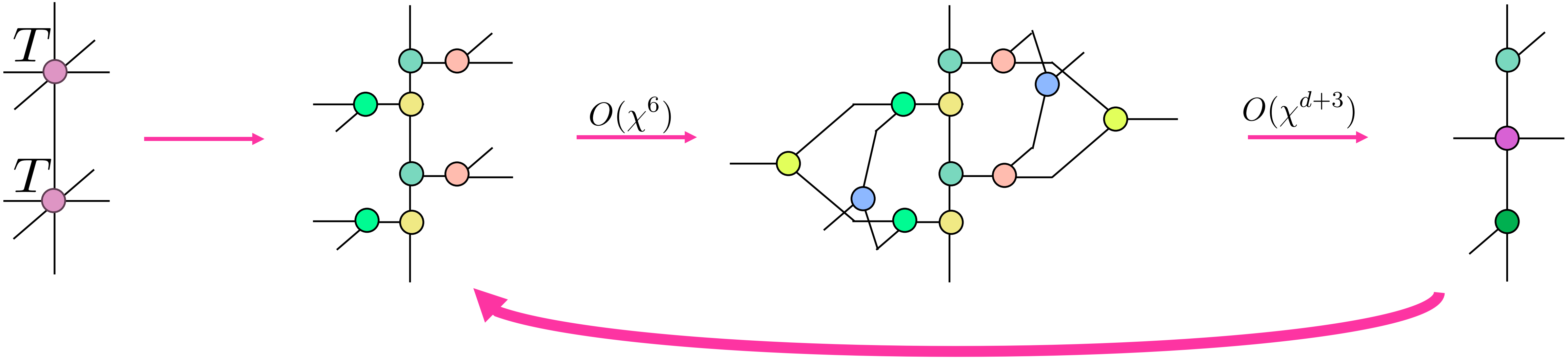
Cost of Triad TRG is. χ^{d+3} !

Two reason of cost reduction

✓ Fundamental tensor has 3 legs

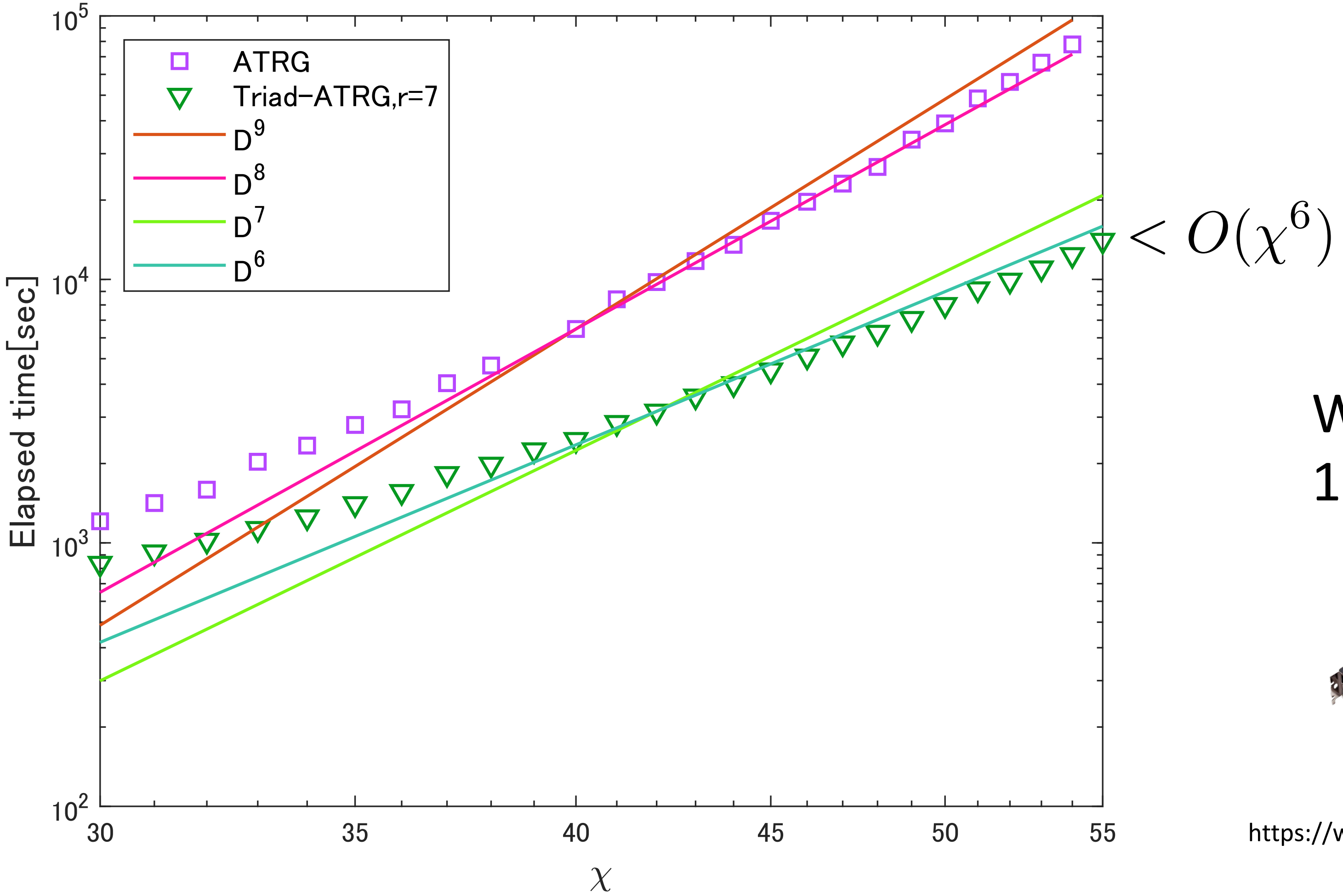
✓ Using RSVD in contraction step

But Triad TRG is less accurate than HOTRG, due to the additional decomposion



COMPUTATIONAL TIME ON GPUS

- We investigate the computational time in r=7 ,L=1024by 2 GPU parallelized calculation
- Scaling of the computational time improved significantly



We have used Tesla V100
16GB PCIE✕2

https://www.elsa-jp.co.jp/wp-content/uploads/2019/03/nvidia_tesla_v100_3qtr.png

HOTRG

HOTRG decomposes subnetwork TT unlike original TRG, which decomposes T

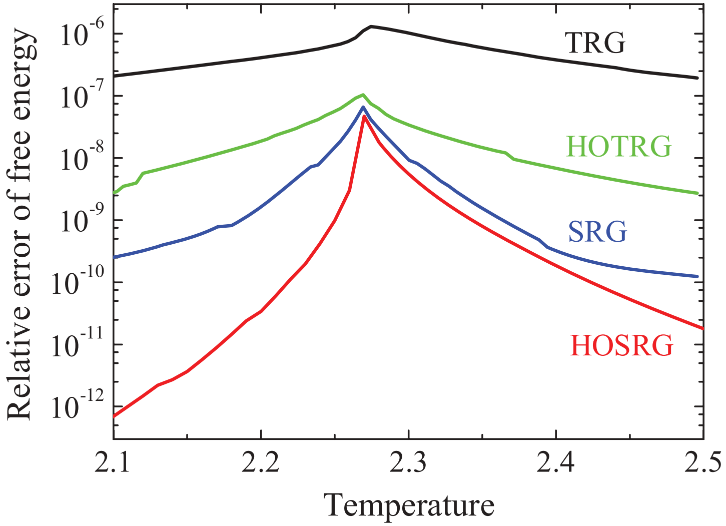


FIG. 4. (Color online) Comparison of the relative errors of free energy with respect to the exact results for the 2D Ising model obtained by various methods with $D = 24$. The critical temperature $T_c = 2/\ln(1 + \sqrt{2})$.

HOTRG has small error from the exact solution.

-Xie et al (2012)

Decomposing large network reduces the systematic error!

BEST LOW RANK APPROXIMATION

Defnition Let $A \in \mathbb{R}^{m\times n}$ The norm of A is defined by

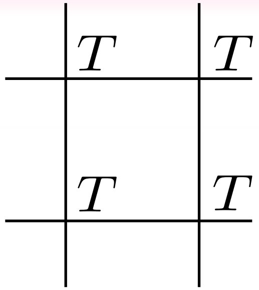
$$||A||_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2} \tag{1}$$

This norm satisfies the following properties.

$$||A||_F^2 = \mathrm{tr}[A^\dagger A] = \sum_{i=1}^{r}\sigma_i^2 \tag{2}$$

TENSOR RENORMALIZATION GROUP

Consider Network



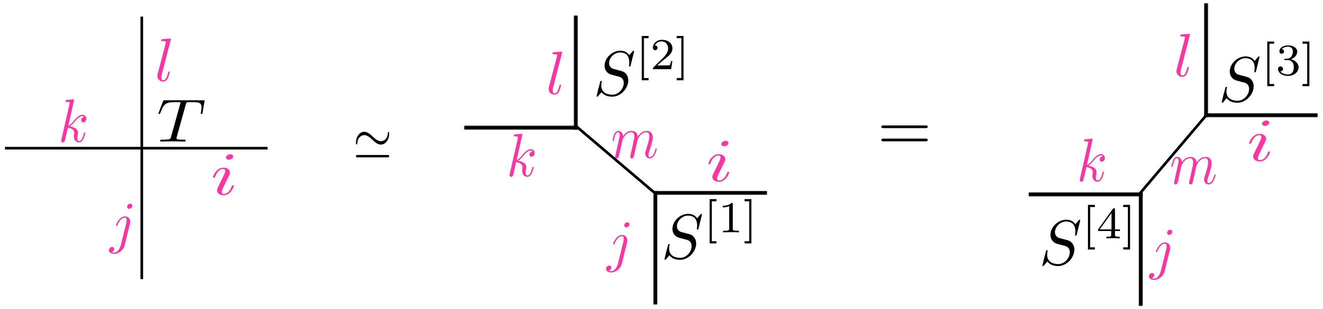
(subnetwork)

Apply SVD for Tensor $T \in \mathbb{C}^{\chi\times\chi\times\chi\times\chi}$ (χ is called the bond dimension, often take 50~100)

$$T_{ijkl} = \sum_{m}^{\chi^2} U_{(ij)m}\sigma_m V^*_{(kl)m} \overset{\text{truncated}}{\simeq} \sum_{m}^{\chi} U_{(ij)m}\sigma_m V^*_{(kl)m} = \sum_{m}^{\chi} S^{[1]}_{(ij)m} S^{[2]}_{(kl)m}$$

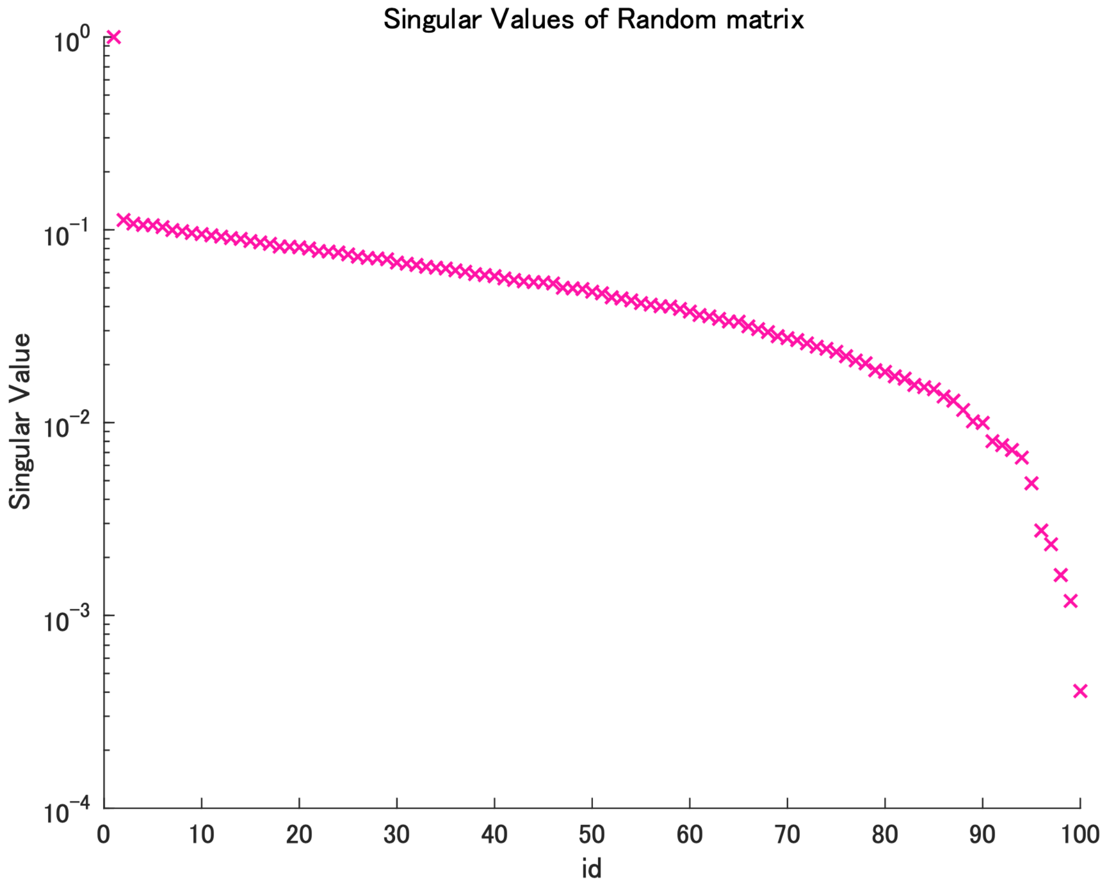
Apply for another pair of indices, we can decompose T below

$T_{ijkl} =$



BEST LOW RANK APPROXIMATION

SVD is the best approximation for matrix. It compress the necessary information of the matrix.



Large singular values is dominant for the norm of the matrix

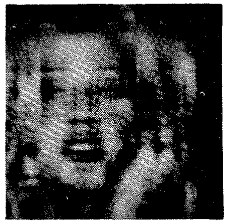
BEST LOW RANK APPROXIMATION

SVD is the best approximation for matrix. It compress the necessary information of the matrix.

128 x 128 pixels

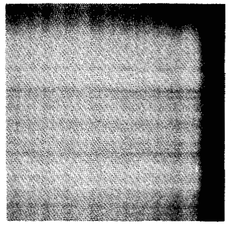
Original image

$$= \sum_{l=1}^{r} \sqrt{\lambda_l}\,\mathbf{u}_l \mathbf{v}_l^T$$



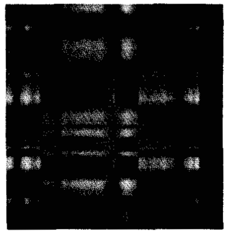
compressed image

$$= \sum_{l=1}^{6} \sqrt{\lambda_l}\,\mathbf{u}_l \mathbf{v}_l^T =$$



$\mathbf{u_1 v_1}^T$

$+ \cdot \quad \cdot +$

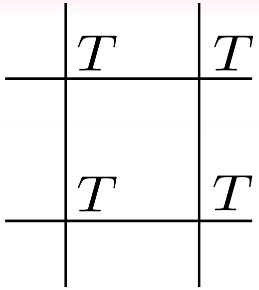


$\mathbf{u_6 v_6}^T$

Using SVD for tensor contraction!

TENSOR RENORMALIZATION GROUP

Consider Network



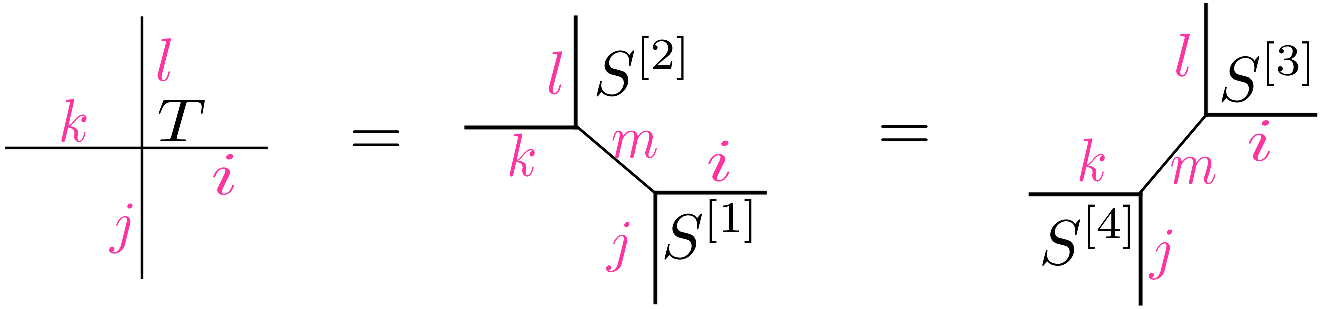
(subnetwork)

Apply SVD for Tensor $T \in \mathbb{C}^{D\times D\times D\times D}$ (D is called the bond dimension, often take 50~100)

$$T_{ijkl} = \sum_{m}^{D^2} U_{(ij)m}\sigma_m V^*_{(kl)m} \overset{\text{truncated}}{\simeq} \sum_{m}^{D} U_{(ij)m}\sigma_m V^*_{(kl)m} = \sum_{m}^{D} S^{[1]}_{(ij)m} S^{[2]}_{(kl)m}$$

Apply for another pair of indices, we can decompose T below

$T_{ijkl} =$

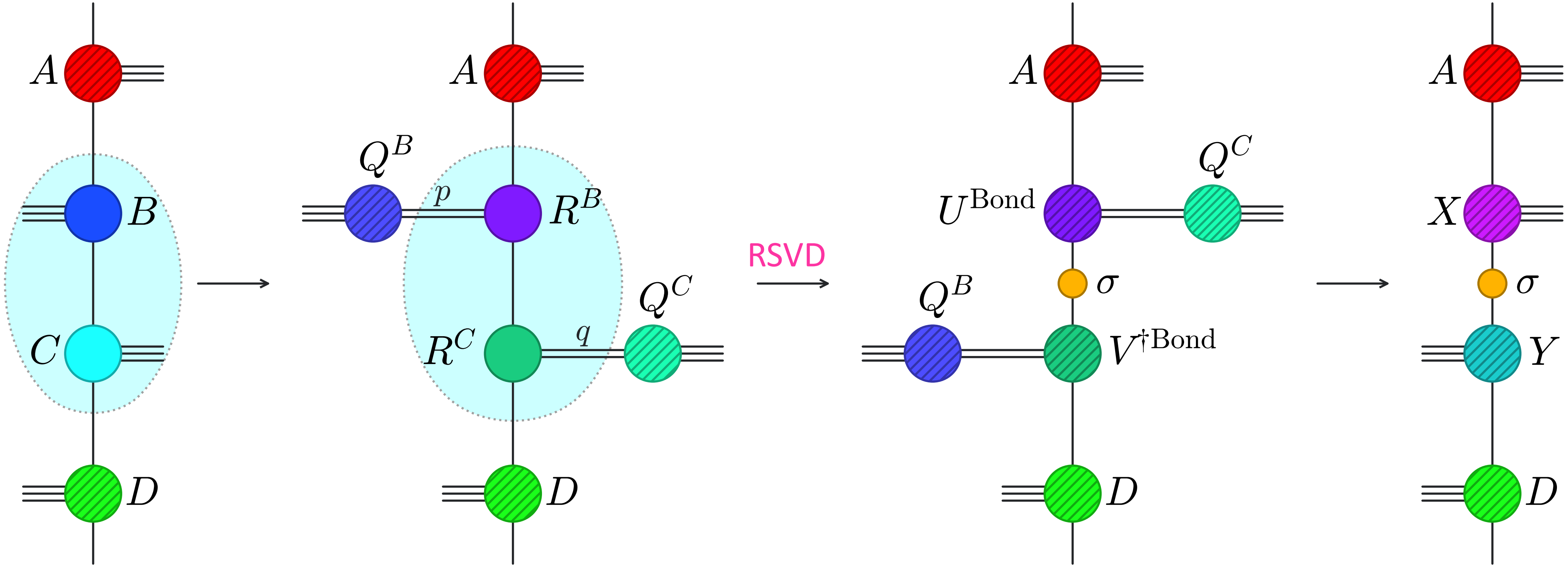


BOND-SWAPPING

Two reason of cost reduction

[D. Adachi, T. Okubo, and S. Todo, (2020).]

- ✓ Fundamental tensor has less legs
- ✓ By doing bond swapping, the # of isometry is reduced to ½



A
B
C
D
A
Q^B
p
R^B
R^C
q
Q^C
D
RSVD
A
Q^C
U^{Bond}
σ
Q^B
$V^{\dagger\mathrm{Bond}}$
D
A
X
σ
Y
D