Workshop: Tensor network 2024 (Kanazawa)

Tensor tree learns hidden relational structure in data to construct generative models

Nov. 16, 2024

Graduate School of Informatics, Kyoto University
Kenji Harada

Collaborators: **Tsuyoshi Okubo** (Univ. of Tokyo), **Naoki Kawashima** (ISSP)

Reference: arXiv:2408.10669

This study is supported by **TOYOTA**

Applications of TN techniques to machine learning

Tensor decomposition in machine learning algorithms

- Tensor train (= **MPS**: matrix product state)

  I.V. Oseledets, "Tensor-Train Decomposition," (2011).

- Compression of tensor in machine learning algorithms

  E. Stoudenmire, and D.J. Schwab, "Supervised Learning with Tensor Networks," (2016).

- Quantics tensor train

$$x = x_N \times 2^{N-1} + x_{N-1} \times 2^{N-2} + \cdots + x_2 \times 2 + x_1$$

Tensorization ➡ $x \to (x_N, x_{N-1}, \cdots, x_1)$

  - Images: Latorre (2005), ...
  - NN: Novikov, et al. (2015), ...

→ **The number of TN applications rapidly grows in machine learning!**

Generative modeling

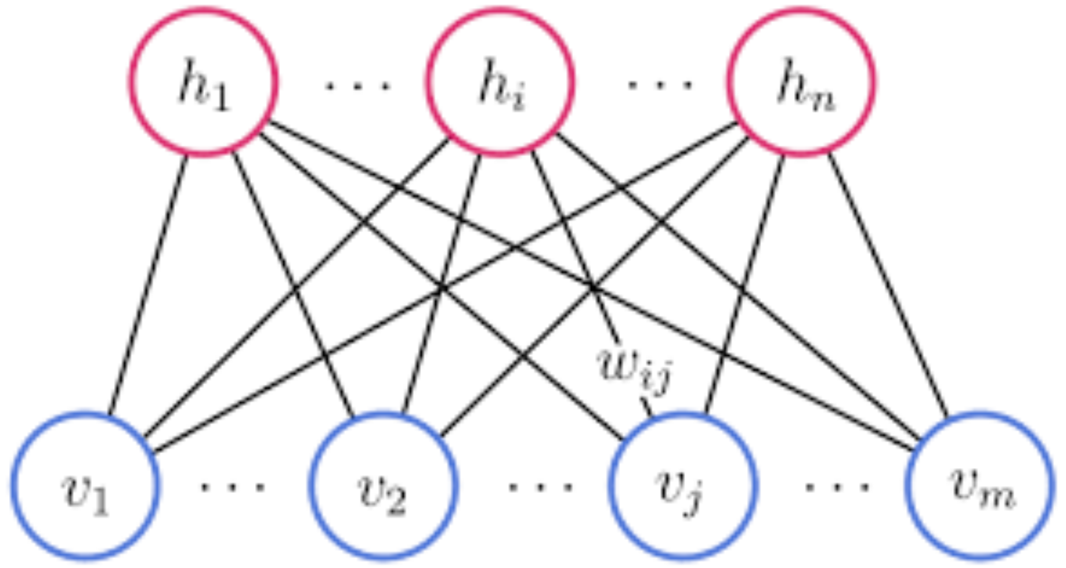
Generative modeling involves creating a classical distribution model behind data.

Learning

$\{\mathbf{x}_i\}_{i\in\mathcal{M}}$

Samples

$P_\theta(\mathbf{x})$ Model of a distribution with parameters θ



Sampling

Generative modeling is an important technique in machine learning

Models for generative modeling

Models for generative modeling

- **Boltzmann** machine and restricted Boltzmann machine (RBM)
- Variational autoencoder (VAE)
- Generative adversarial network (GAN)
- Normalizing flow
- Diffusion model

Magnet (spin)

Based on classical physics

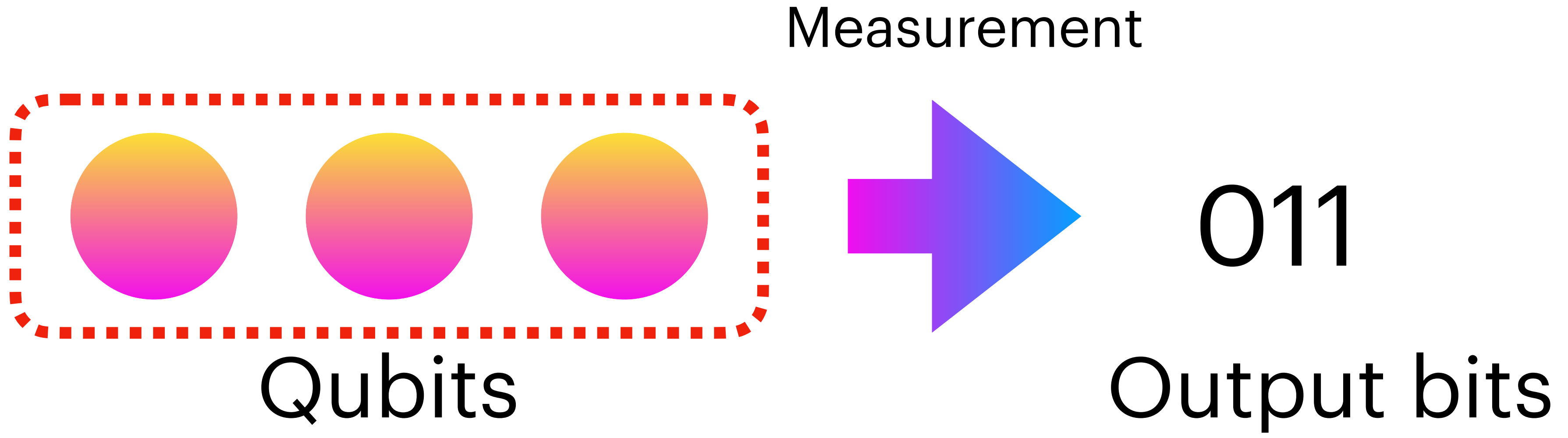
Diffusion process

Born machine

Based on projective measurements of a **quantum** state

$$p(\mathbf{x}) = |\psi(\mathbf{x})|^2$$

Z.-Y. Han, J. Wang, H. Fan, L. Wang, and P. Zhang, Phys. Rev. X **8**, 031012 (2018).



$$|\psi\rangle = \cdots + c_{011}|011\rangle + \cdots$$

$$P(011) = |c_{011}|^2$$

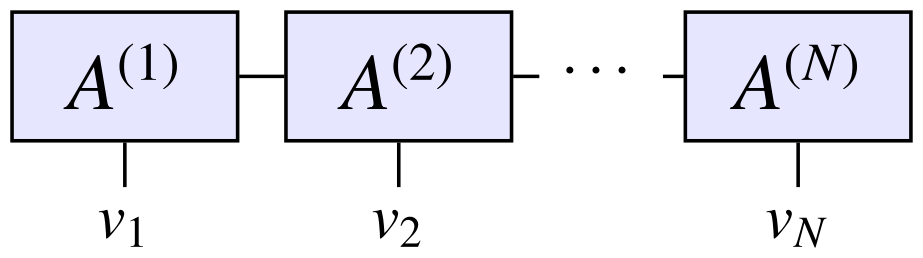
Two approaches for the Born machine

Tensor network model (TN)

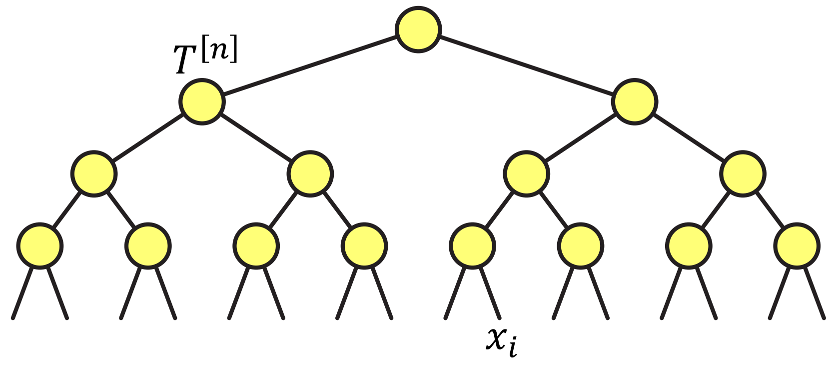
- Zhao-Yu Han, Jun Wang, Heng Fan, Lei Wang, and Pan Zhang, “Unsupervised Generative Modeling Using Matrix Product States,” Physical Review X, **8**, 031012(2018).
- Song Cheng, Lei Wang, T. Xiang, and Pan Zhang, “Tree tensor networks for generative modeling,” Physical Review B, **99**, 155131(2019).

$$|\psi(x)\rangle \equiv$$

MPS



Binary tree-TN



MERA, PEPS, ...

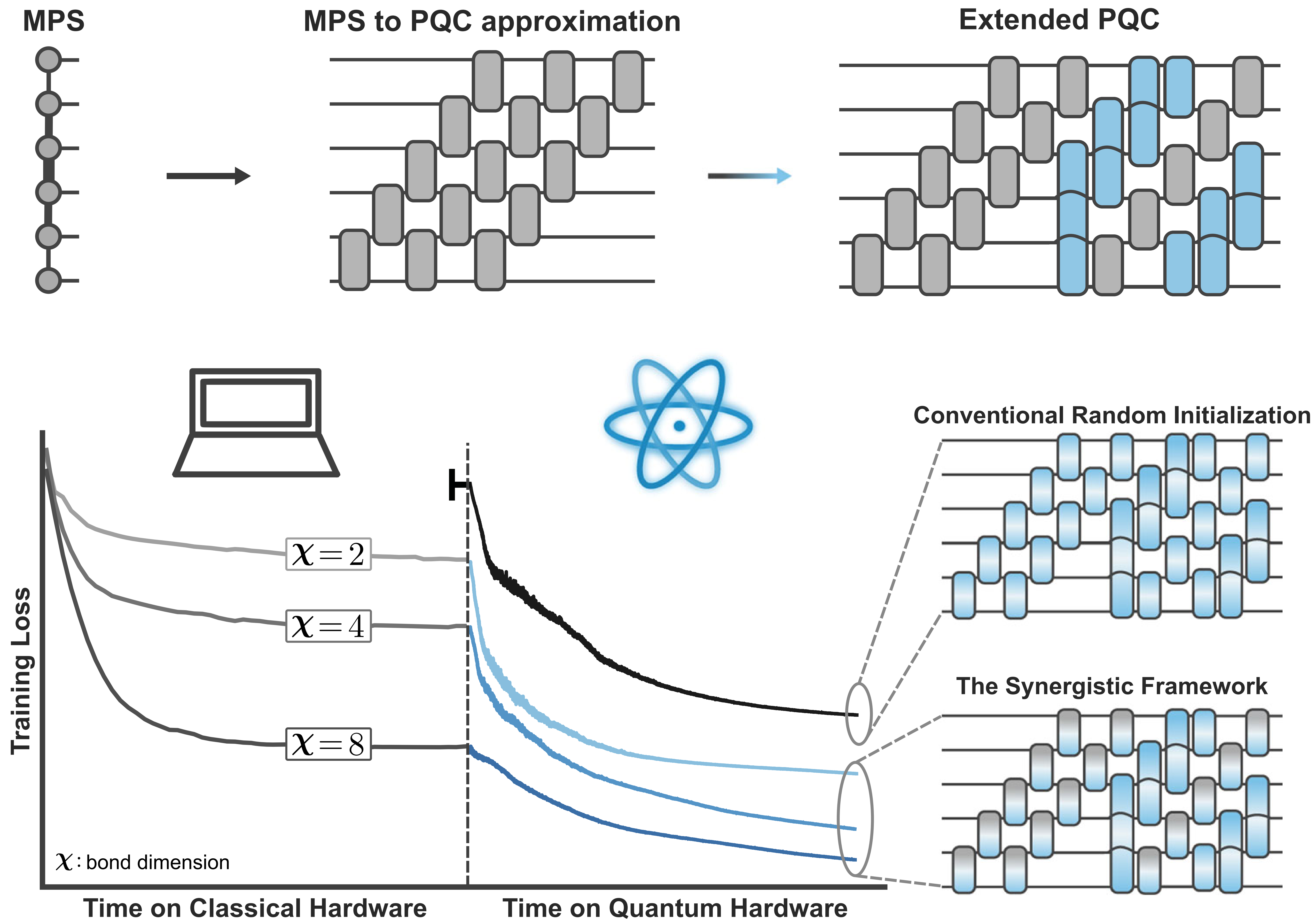
Parametrized quantum circuit model (PQC)

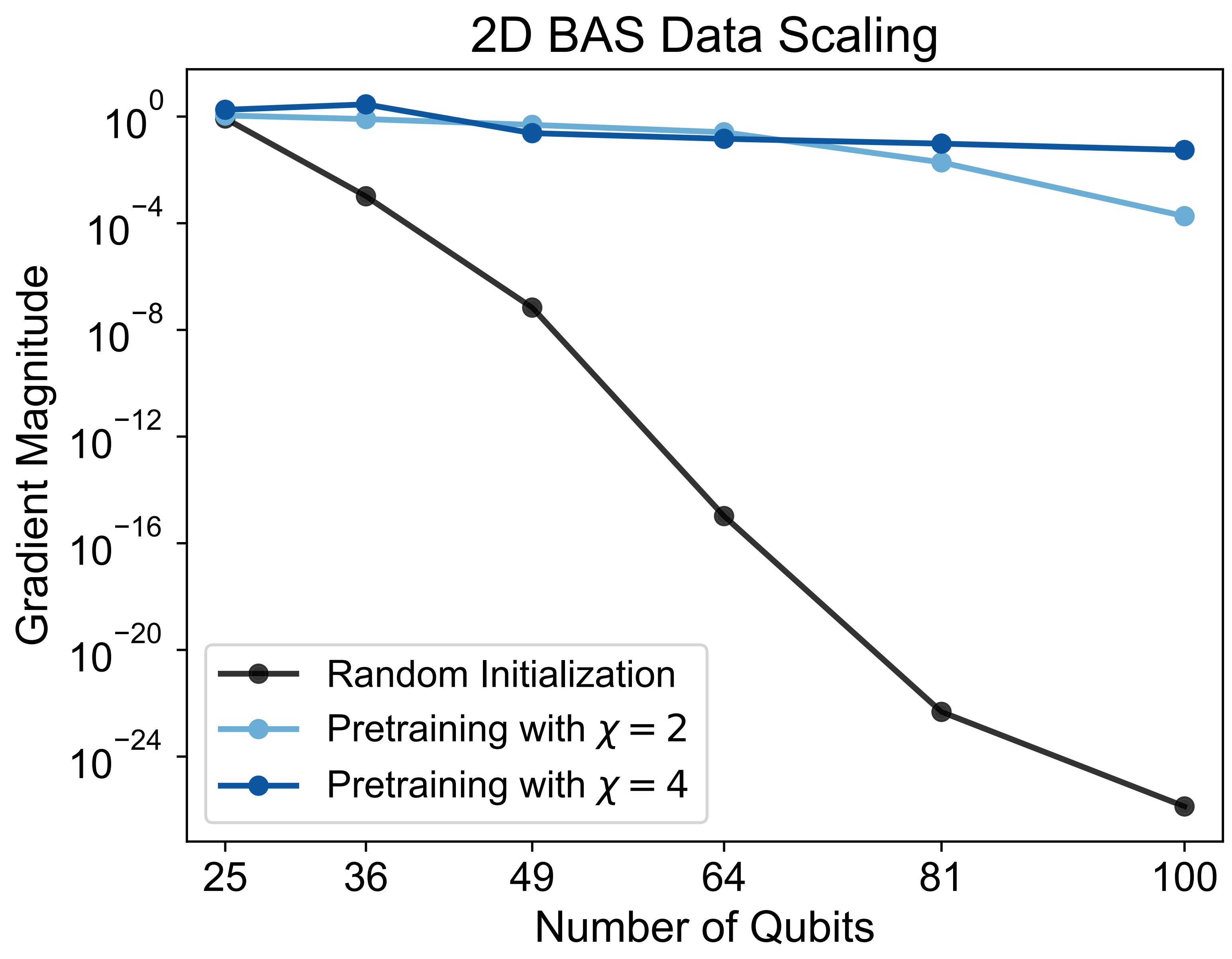
- Jin-Guo Liu and Lei Wang, “Differentiable learning of quantum circuit Born machines,” Physical Review A, **98**, 062324(2018).
- Marcello Benedetti, *et al.*, “A generative modeling approach for benchmarking and training shallow quantum circuits,” npj Quantum Information, **5**, 45(2019).
- Brian Coyle, et al., “The Born supremacy: quantum advantage and training of an Ising Born machine,” npj Quantum Information, **6**, 60(2020).
- Marcello Benedetti, *et al.*, “Variational Inference with a Quantum Computer,” Physical Review Applied, **16**, 044057(2021).
- Manuel S Rudolph, *et al.*, “Synergistic pretraining of parametrized quantum circuits via tensor networks,” Nature Communications, **14**, 8367(2023).
- Mohamed Hibat-Allah, *et al.*, “A framework for demonstrating practical quantum advantage,” Communications Physics, **7**, 68(2024).

Synergistic pretraining of parametrized quantum circuits via tensor networks

Manuel S Rudolph, et al., Nature Communications, **14**, 8367(2023).

Synergistic approach by TN and PQC





The synergistic approach resolves the issue of the prevalence of barren plateaus in PQC optimization landscapes

Network structure and performance

S. Cheng, L. Wang, T. Xiang, and P. Zhang, Phys. Rev. B, **99**, 155131(2019).

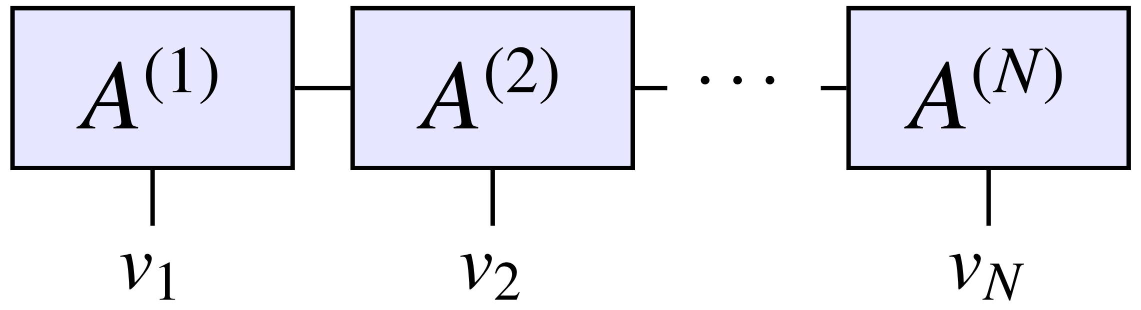
Loss function

Negative Log-likelihood (NLL) = KL-divergence - entropy of data

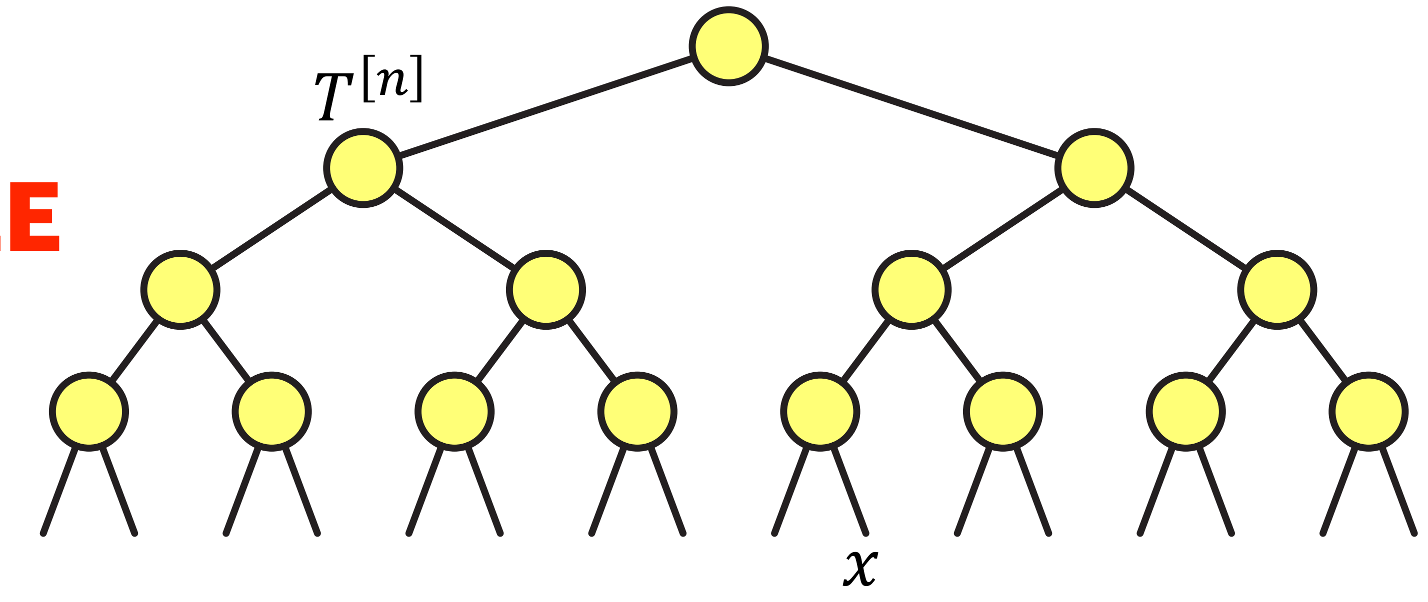
$$\mathcal{L} = -\frac{1}{|\mathcal{T}|}\sum_{\mathbf{x}\in\mathcal{T}} \ln[p(\mathbf{x})] = -\frac{1}{|\mathcal{T}|}\sum_{\mathbf{x}\in\mathcal{T}} \ln|\Psi(\mathbf{x})|^2$$

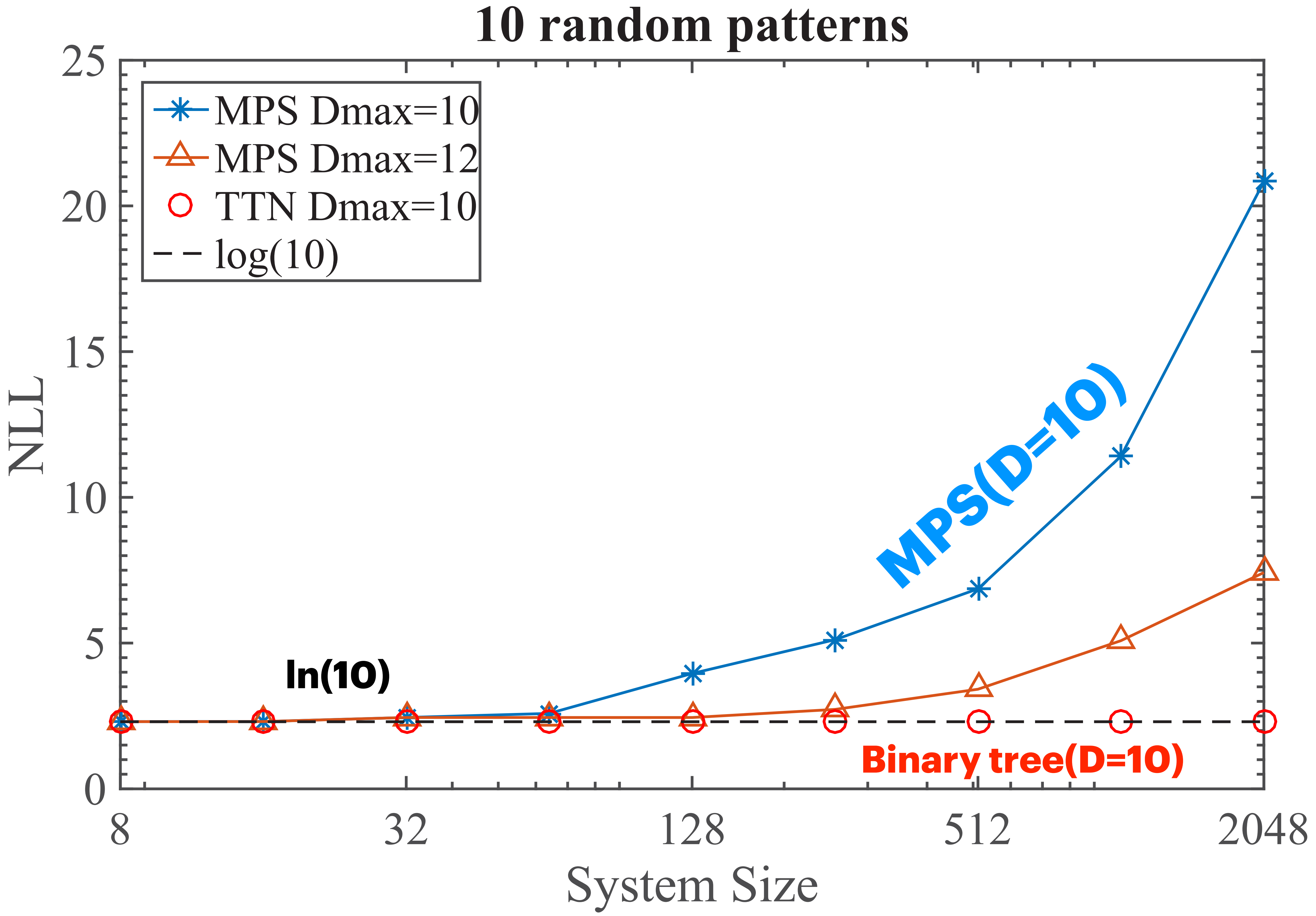
TN of the quantum state for the Borm machine

$\Psi(\mathbf{x}) =$ **MPS**



Binary TREE





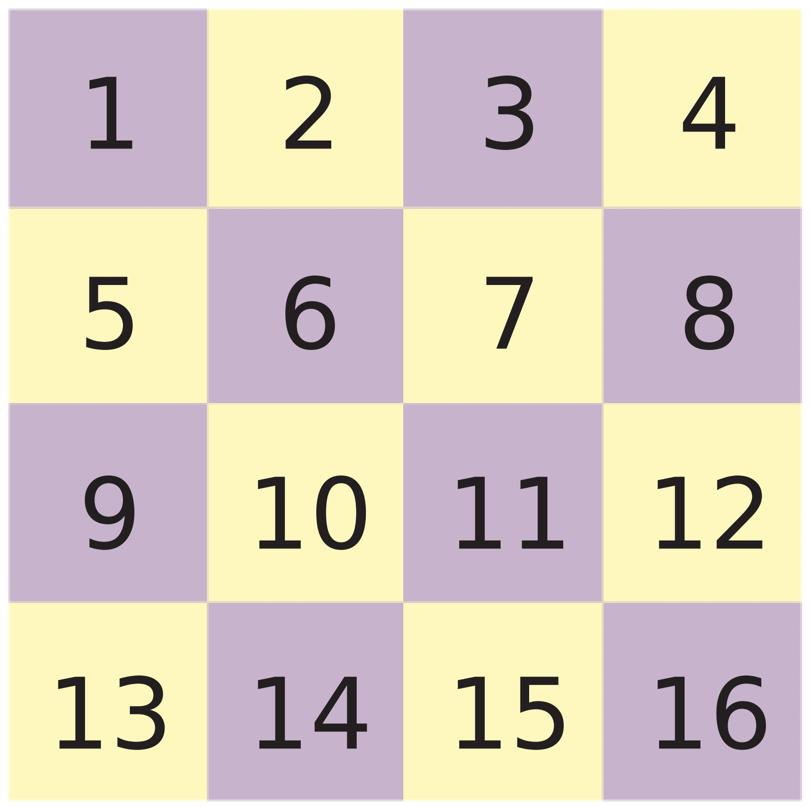
The binary tree reaches the optimal value of NLL.

Tensor network structure with good performance

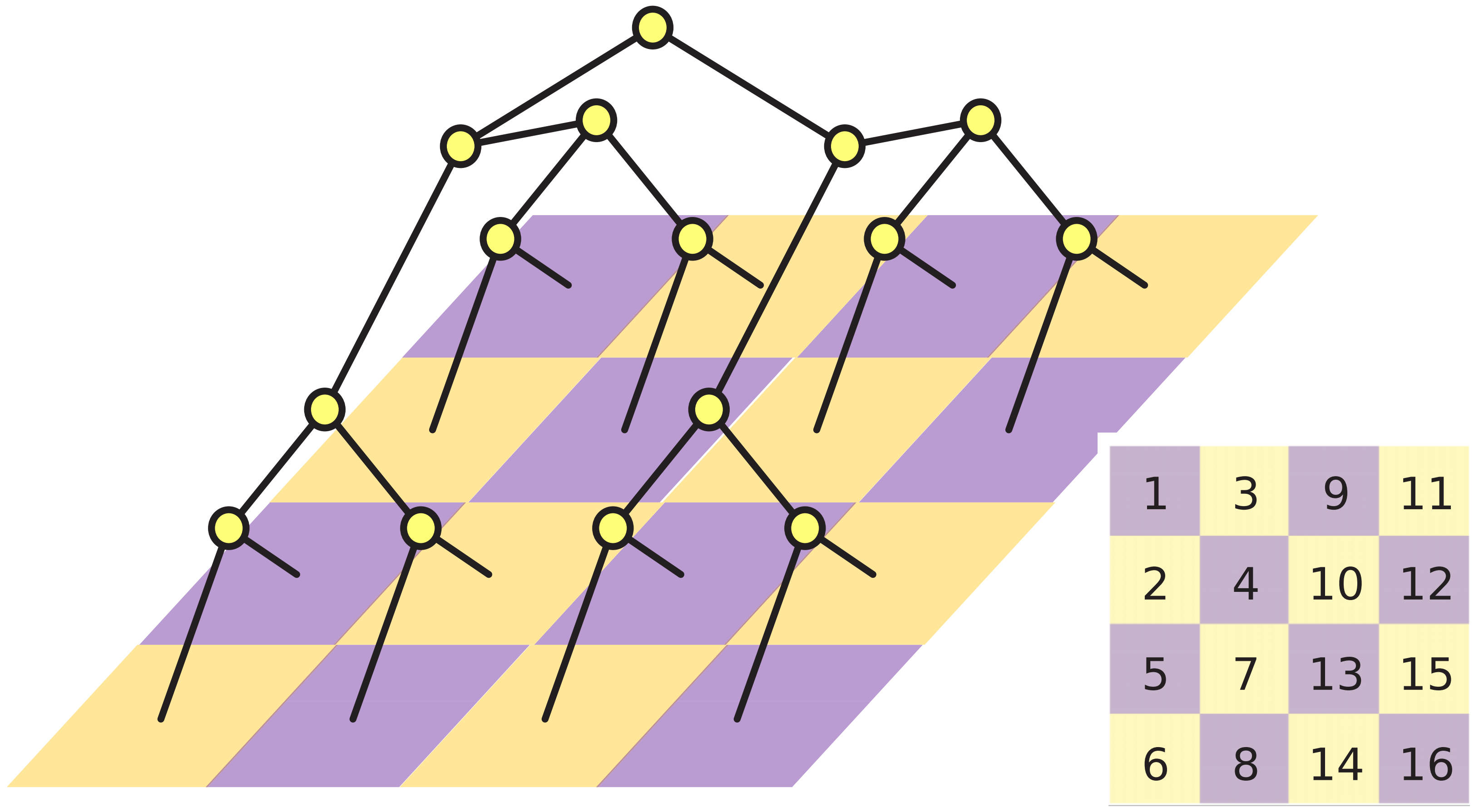
Use of prior knowledge of data

Ex. Images of hand-written digits

Network structures align with **geometrical relationships** among pixels



MPS



Binary tree TN with 2D structure

For **no prior knowledge** of data, how can we design a good network structure?

Optimizing a tensor network structure

Usually, we first fix a tensor network structure.

In the case of the ground-state calculation, the network structure aligns with local interactions on a lattice.

Ex. 1D model ⇒ MPS,   2D model ⇒ PEPS

However, we often have **no prior** knowledge of data.

Our goal

Optimizing a tensor network structure for generative modeling without prior knowledge of data.

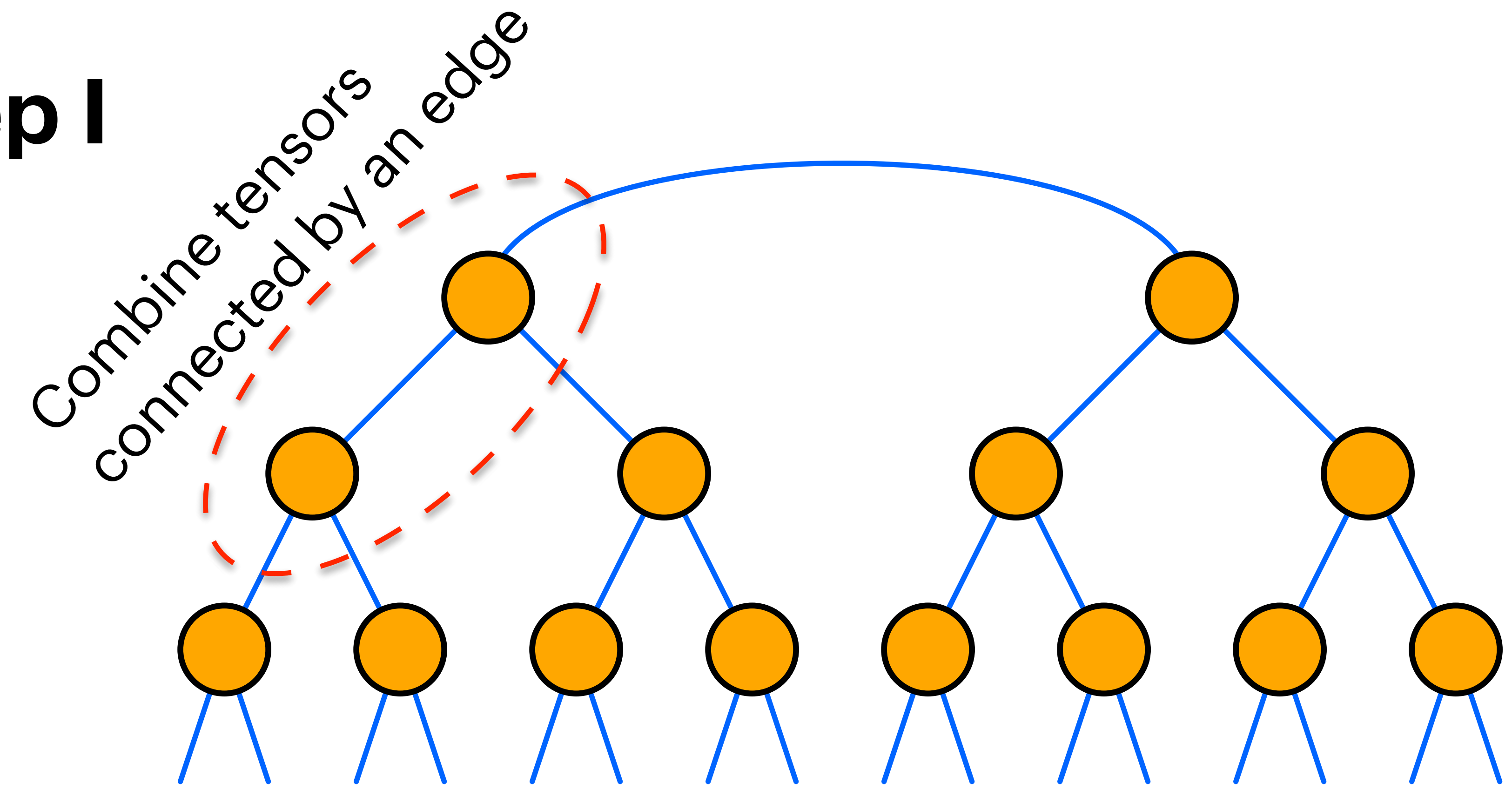
Optimization of network structure for a ground-state calculation

T. Hikihara, H. Ueda, K. Okunishi, **K.H.**, and T. Nishino, Phys. Rev. Research **5**, 013031 (2023)

In the class of **general tree** TNs

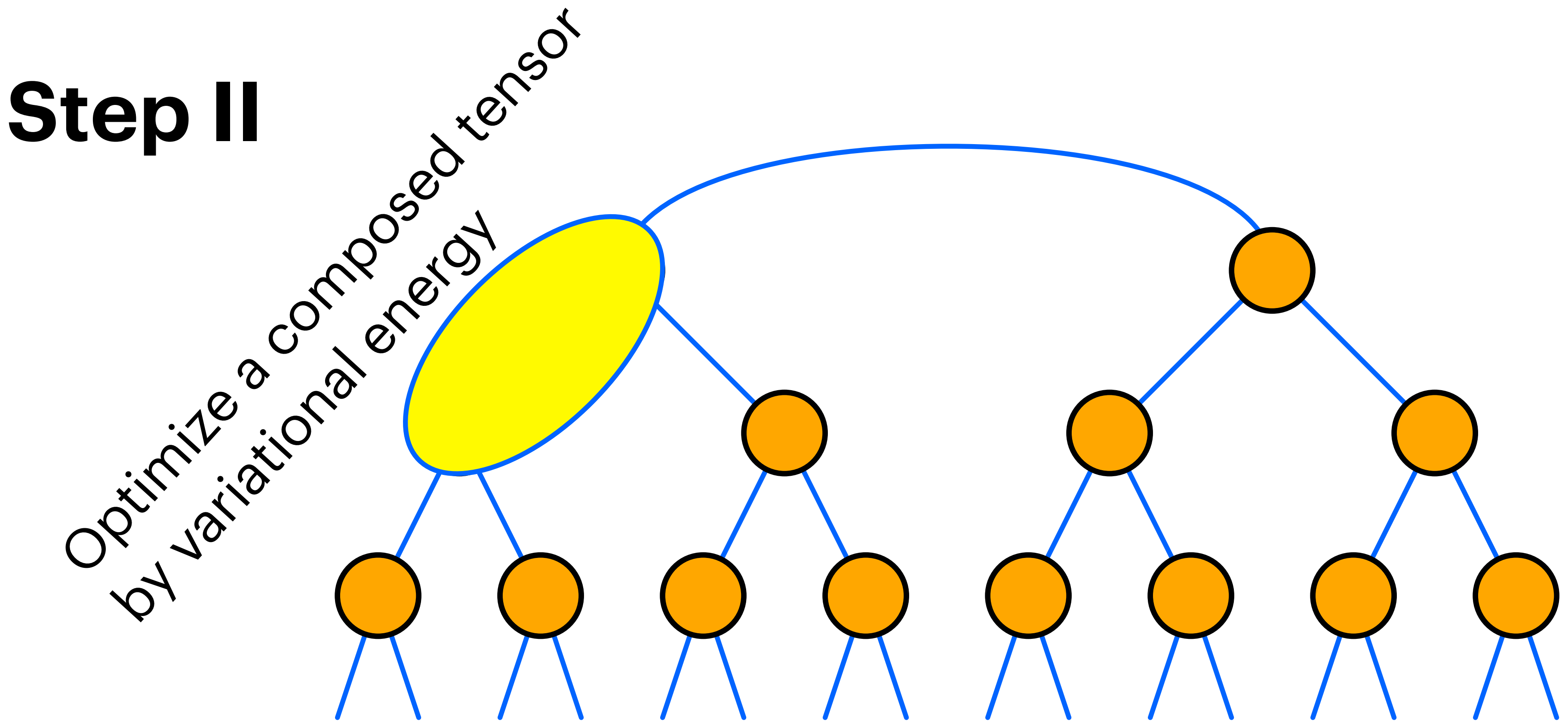
Based on a two-sites algorithm of DMRG

Step I

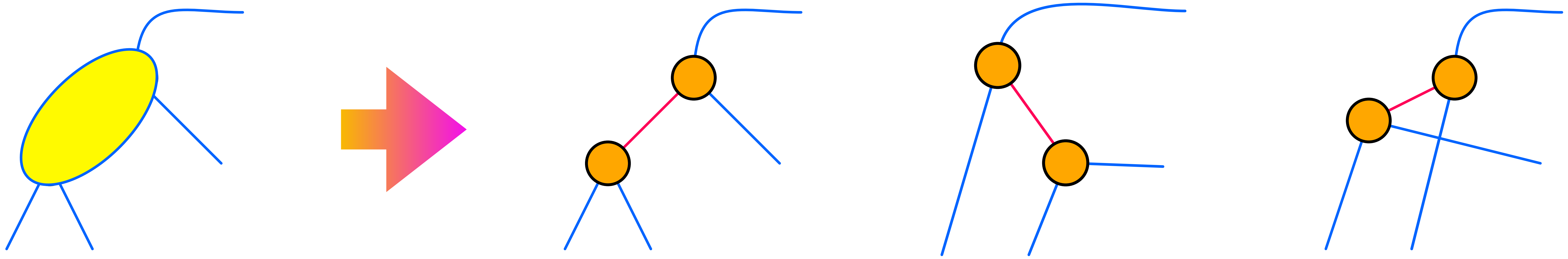


Note. Use of a canonical form

Step II



Step III Select a new decomposition with **the least entanglement entropy**



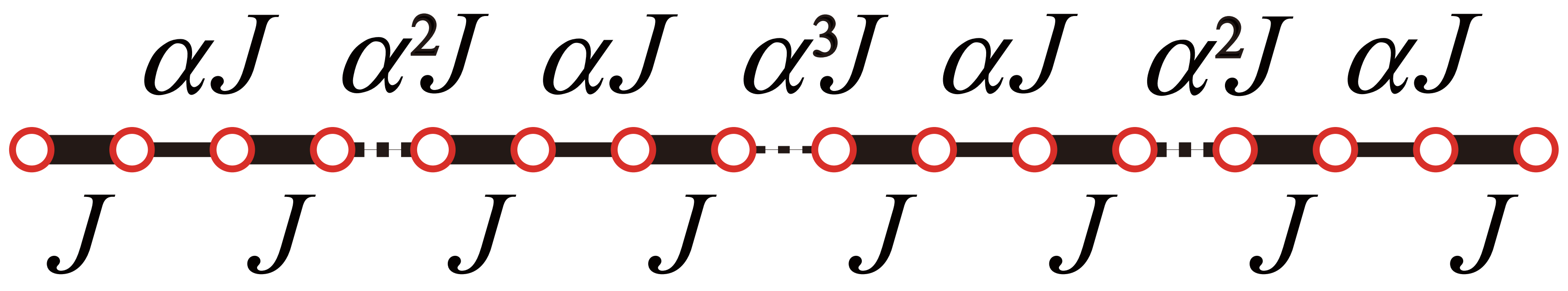
Truncation is small

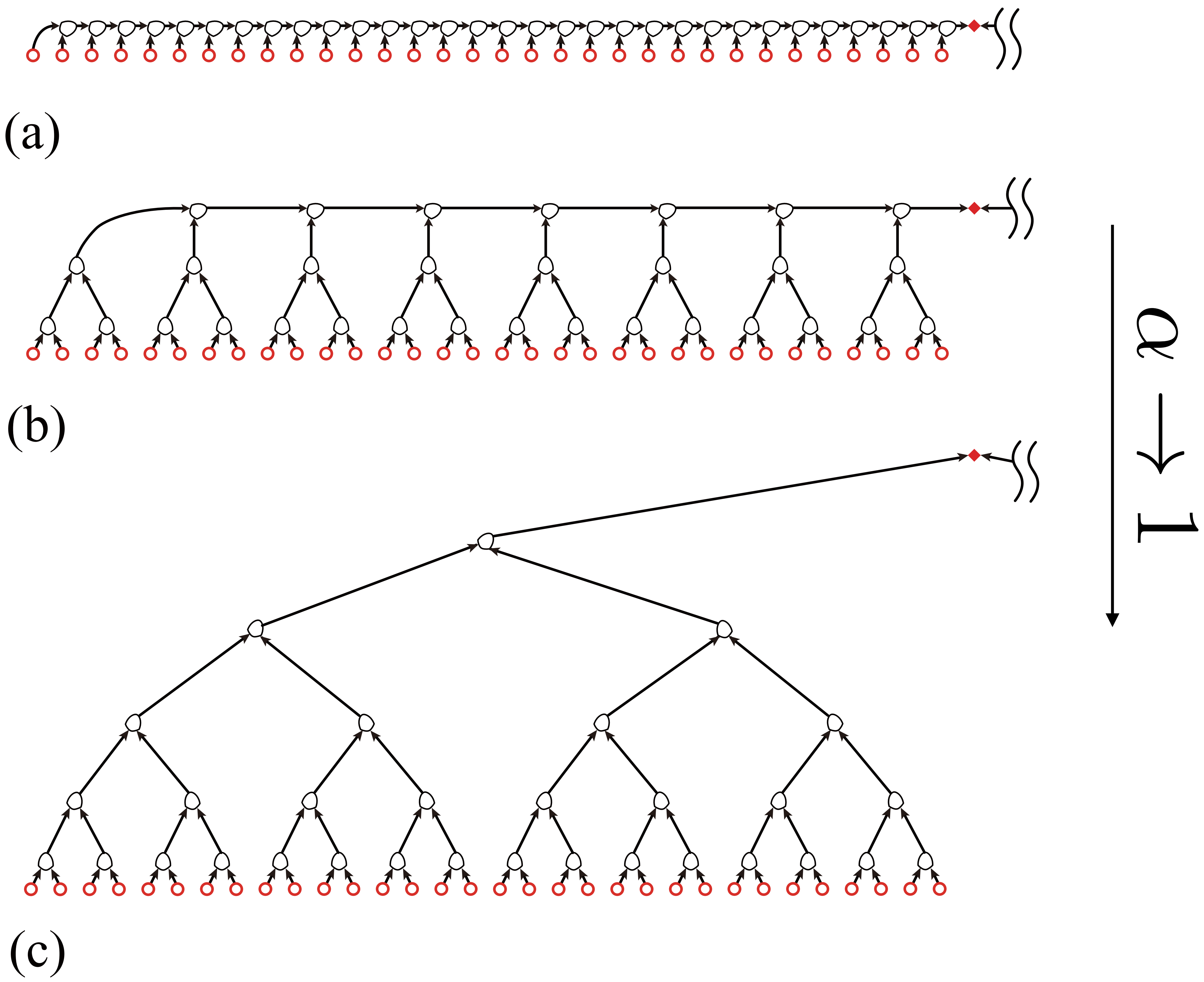
Results of optimization of network structure for a ground-state calculation

T. Hikihara, H. Ueda, K. Okunishi, **K.H.**, and T. Nishino, Phys. Rev. Research **5**, 013031 (2023)

- Visualization of entanglement structure

Inhomogeneous AFH



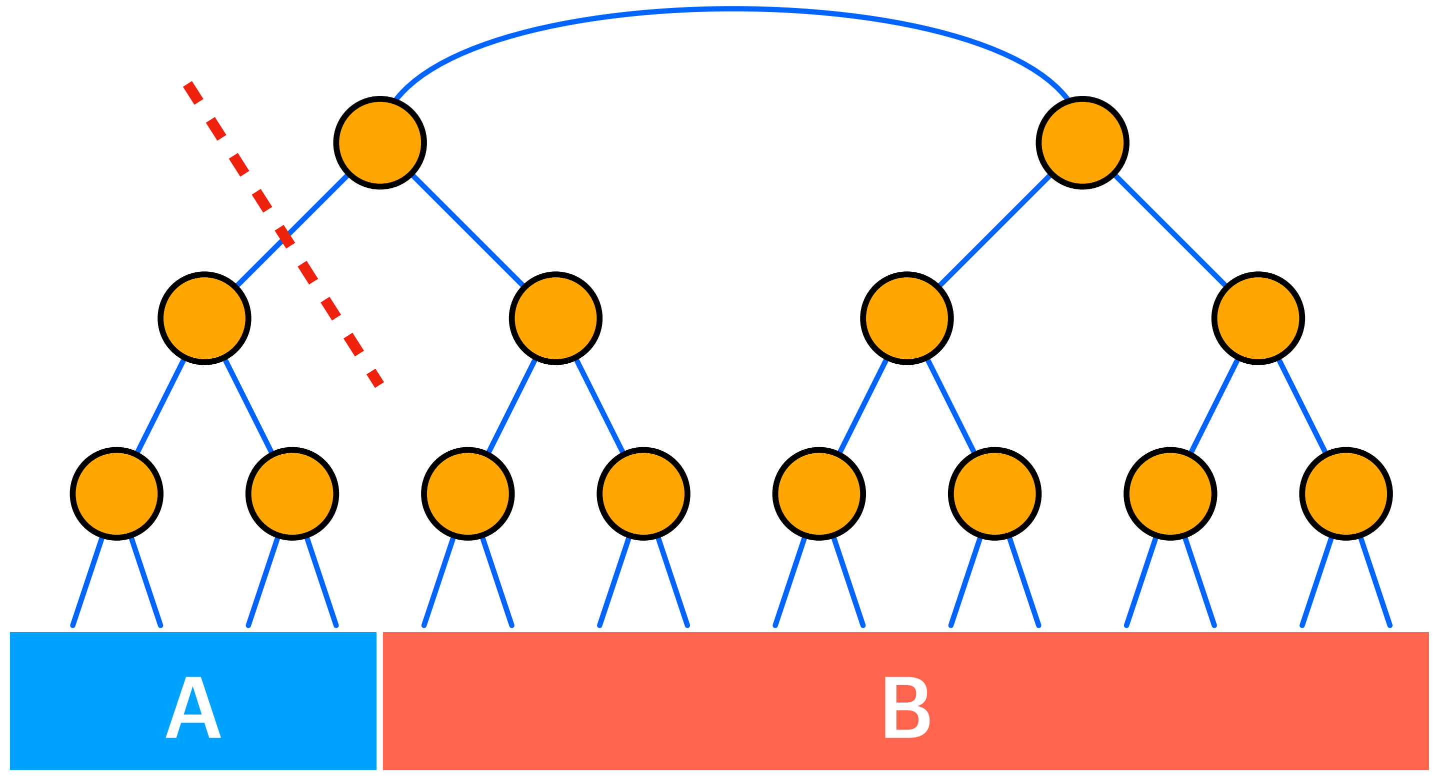


- Improvement of variational energies

Classical mutual information and entanglement

Classical mutual information

$$I(A:B) = \sum_{(a,b)} P(a,b) \ln\left[\frac{P(a,b)}{P(a)P(b)}\right]$$

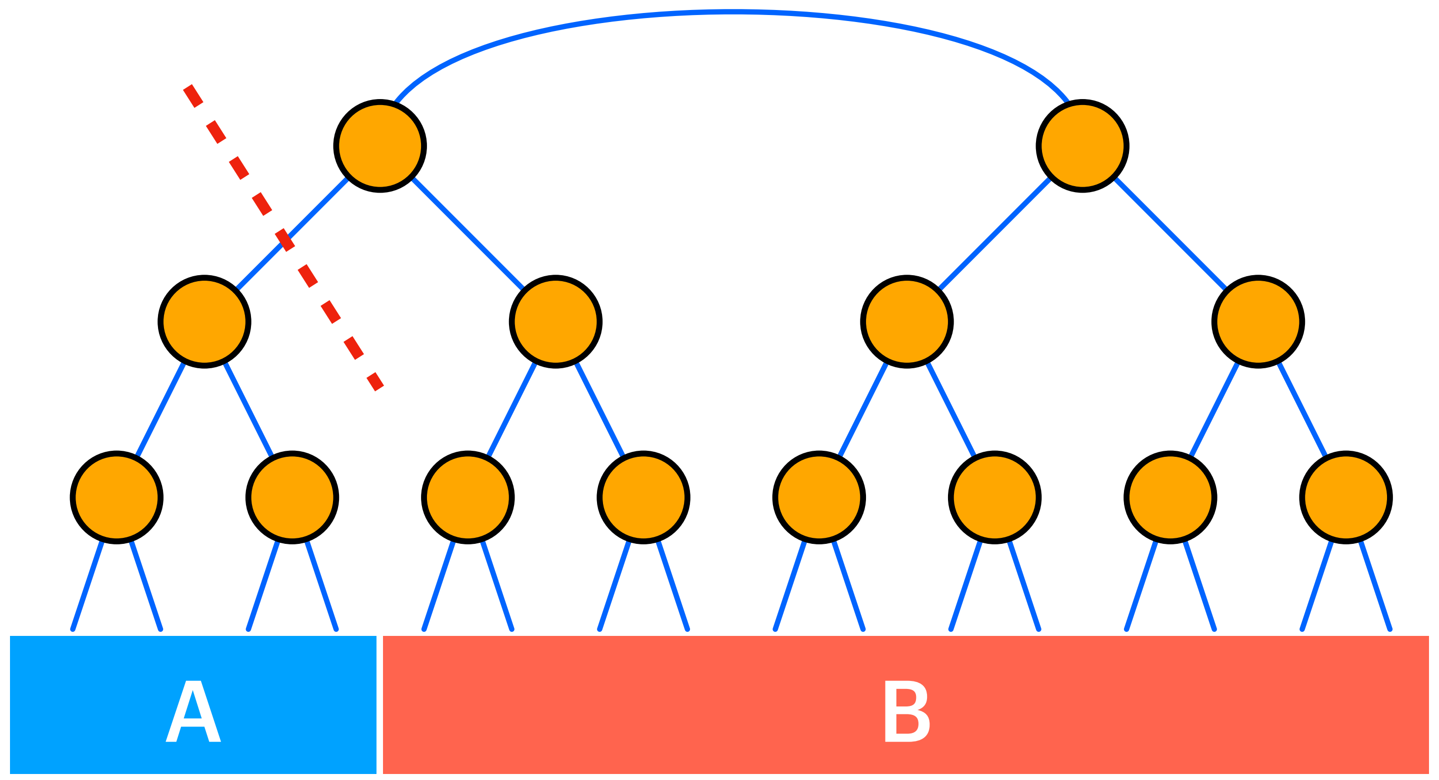


Classical mutual information and entanglement

Classical mutual information

Entanglement entropy

$$I(A:B) = \sum_{(a,b)} P(a,b) \ln \left[\frac{P(a,b)}{P(a)P(b)} \right] \leq \text{E.E.}$$



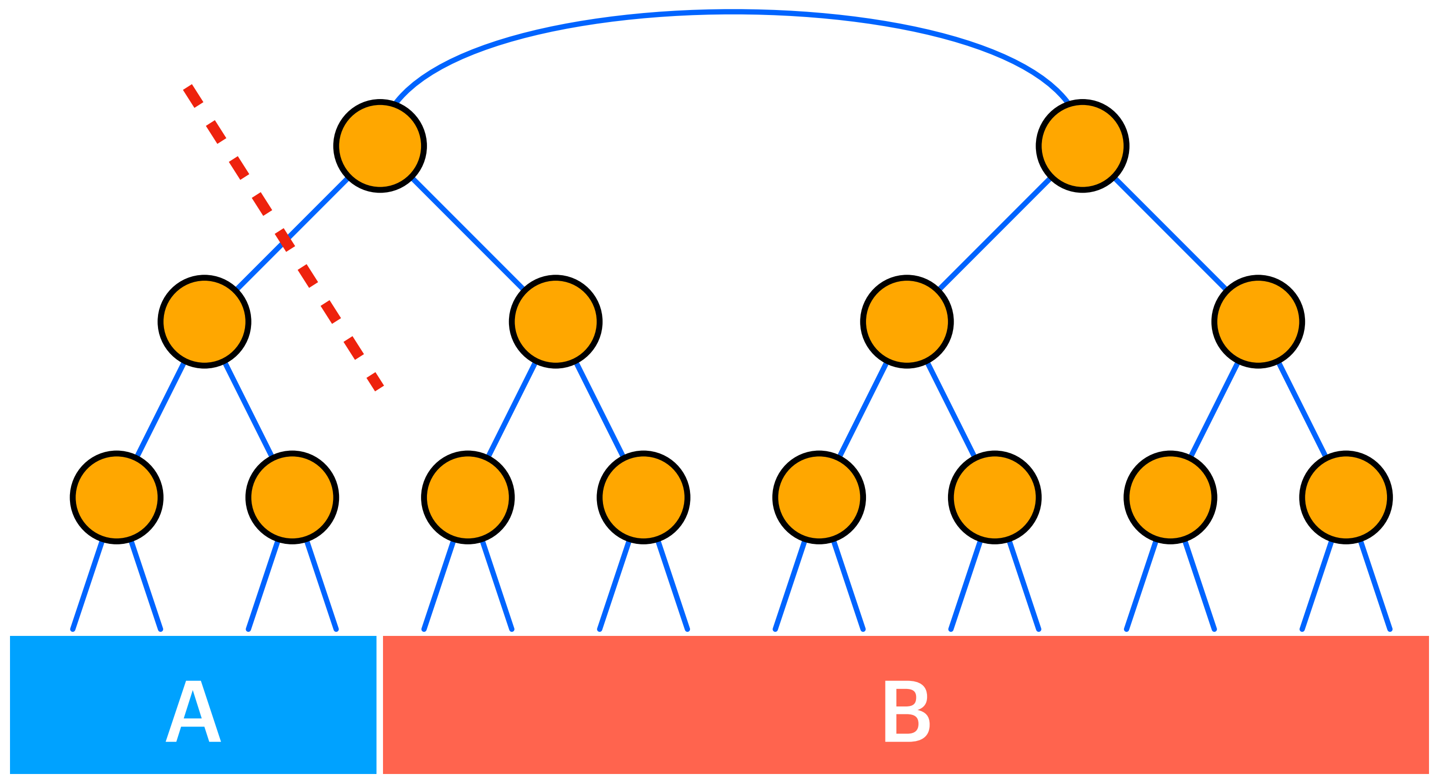
Classical mutual information and entanglement

Classical mutual information

Entanglement entropy

$$I(A:B) = \sum_{(a,b)} P(a,b) \ln\left[\frac{P(a,b)}{P(a)P(b)}\right] \leq \text{E.E.} \leq \ln(D)$$

in a tree



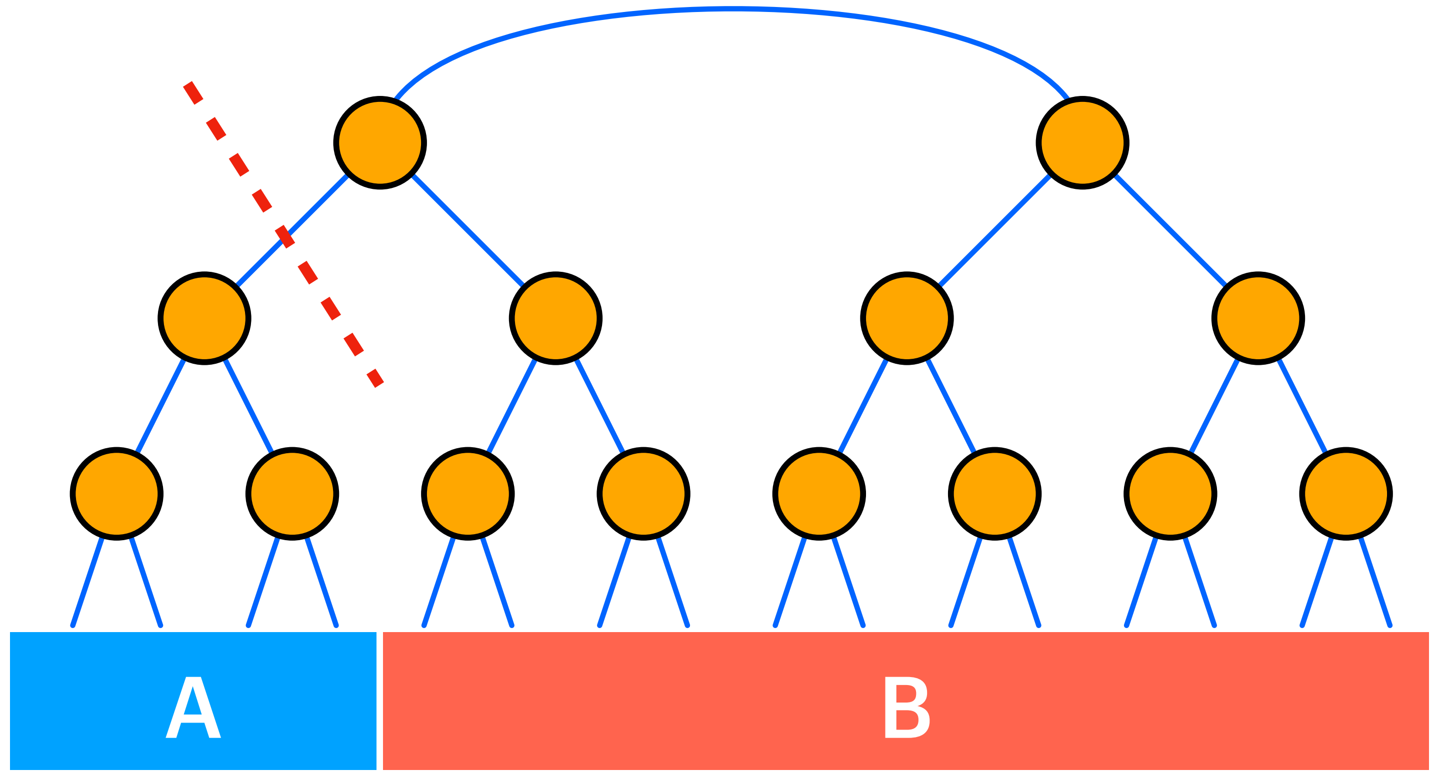
Classical mutual information and entanglement

Classical mutual information

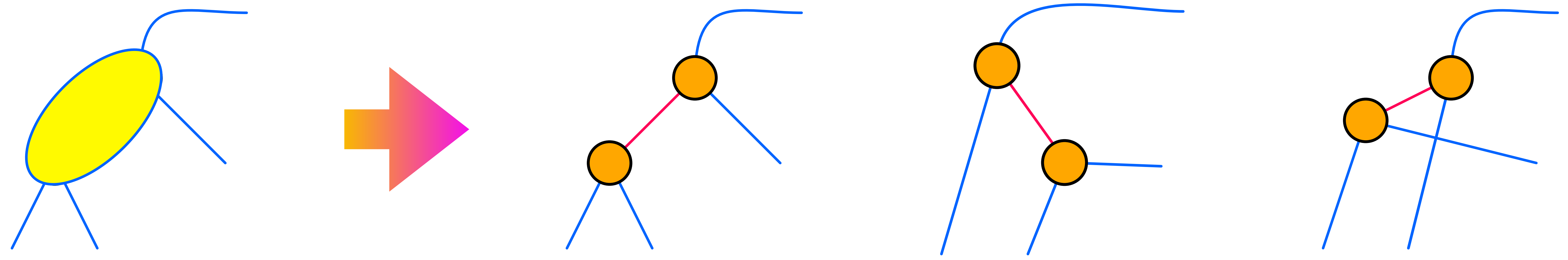
Entanglement entropy

$$I(A:B) = \sum_{(a,b)} P(a,b)\ln\left[\frac{P(a,b)}{P(a)P(b)}\right] \leq \text{E.E.} \leq \ln(D)$$

in a tree



Selecting a new decomposition with **the least** classical mutual information

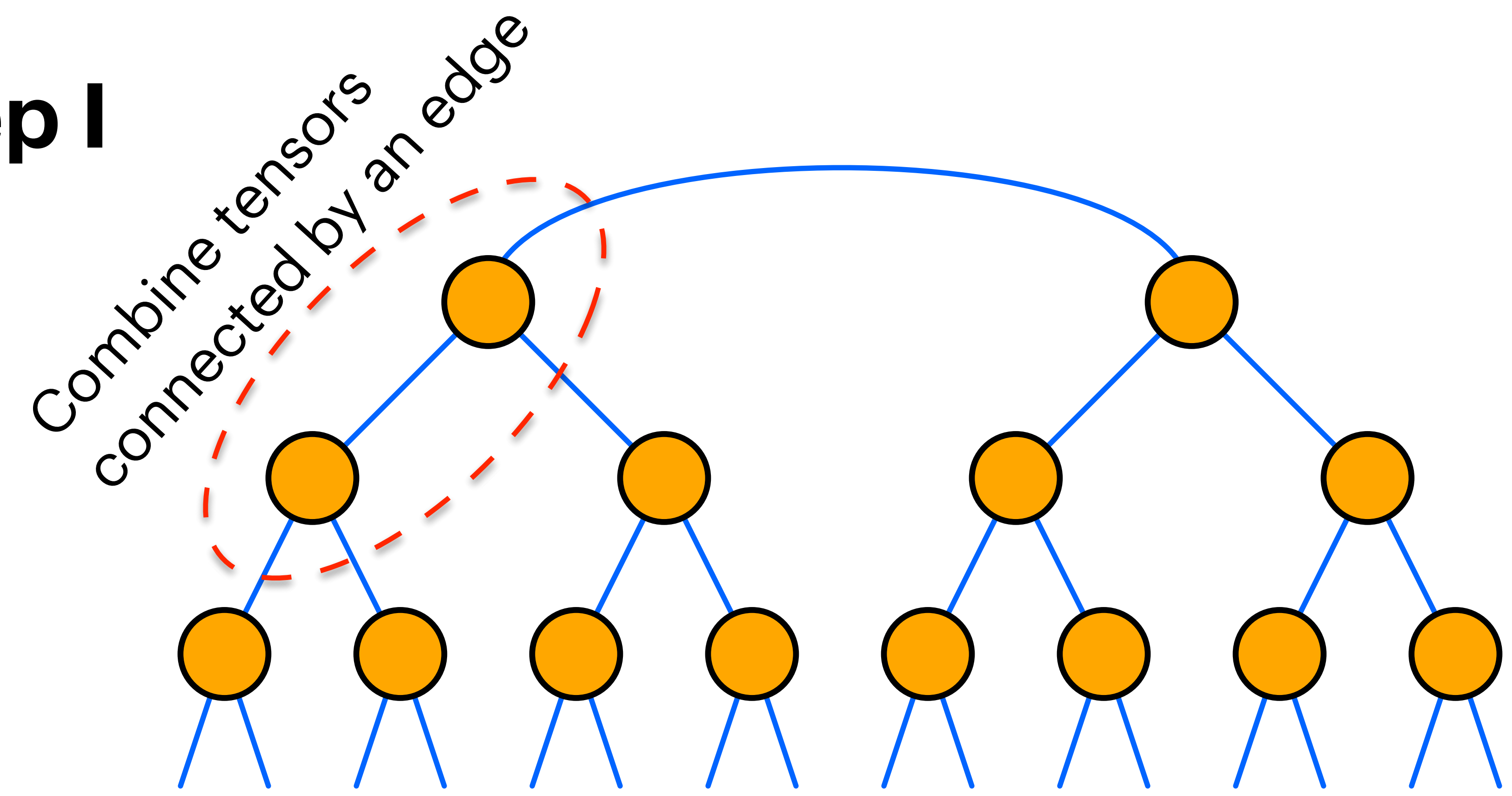


Adaptive tensor tree generative modeling

K.H., Tsuyoshi Okubo and Naoki Kawashima, arXiv:2408.10669

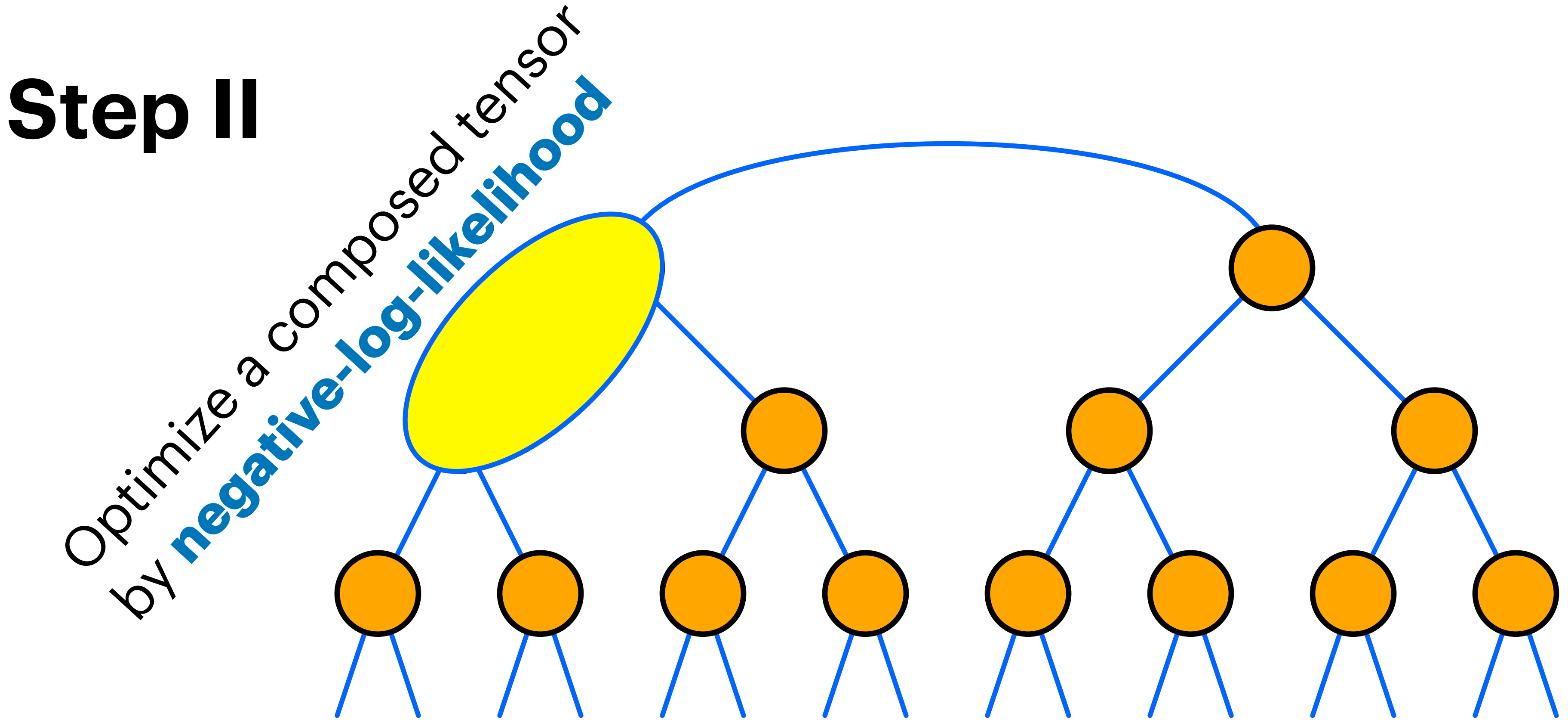
In the class of **general tree** TNs

Step I

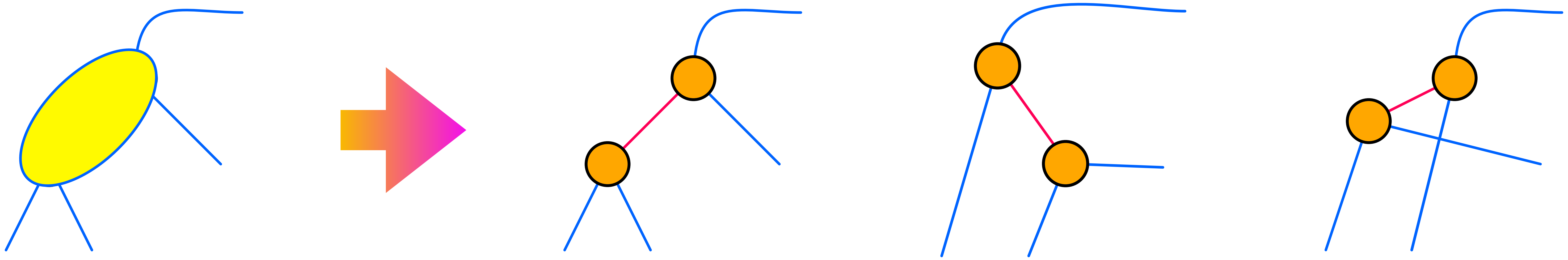


Note. Use of a canonical form

Step II



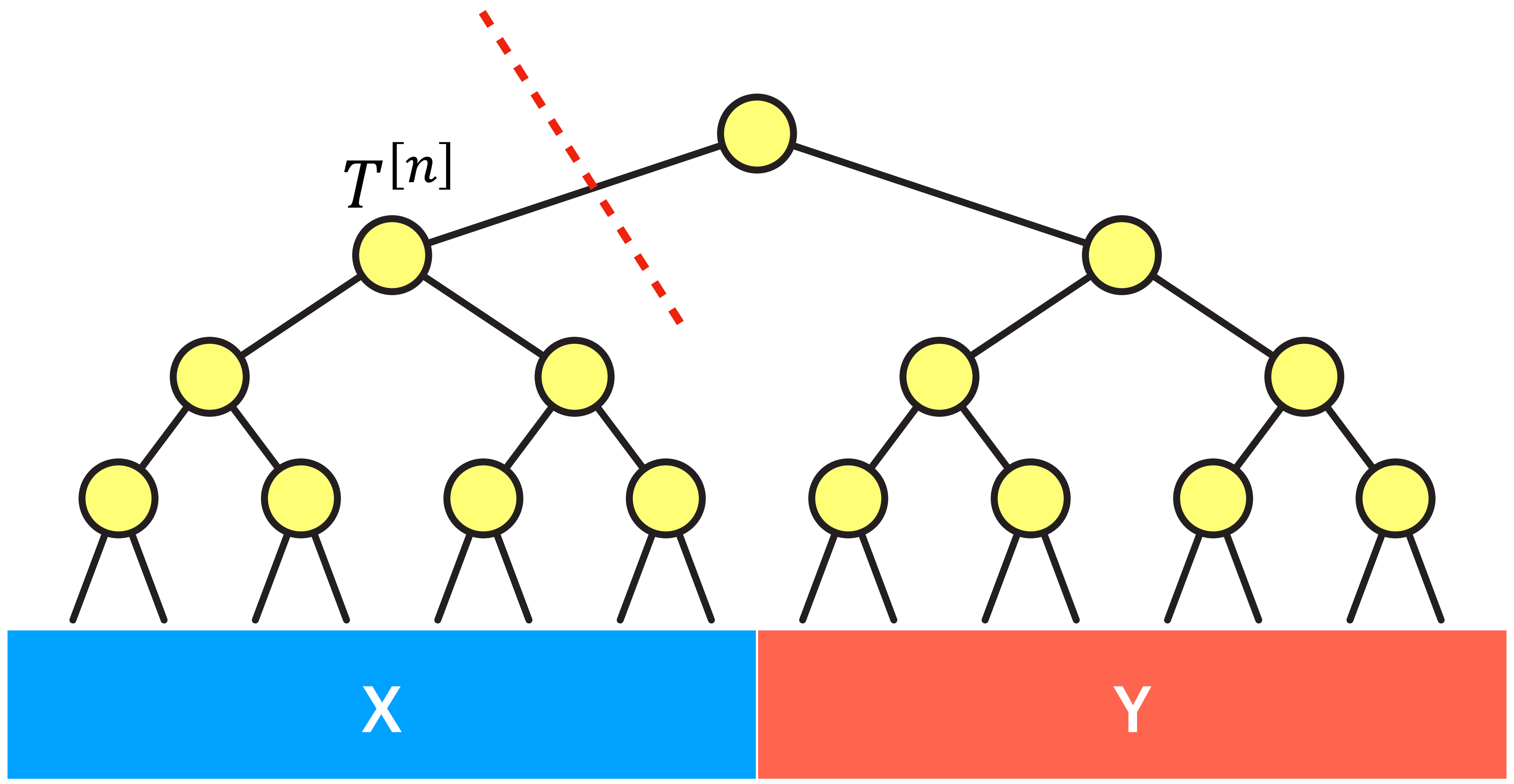
Step III Select a new decomposition with **the least classical mutual information**



Stochastic estimation of mutual information

$$\mathrm{MI} = \sum_{X,Y} P(X,Y) \ln \left[\frac{P(X,Y)}{P(X)P(Y)} \right] \quad \Rightarrow \quad \mathrm{MI} \approx \frac{1}{M} \sum_{\substack{(X_i,Y_i)\in\mathcal{T} \\ 1\le i\le M}} \ln \left[\frac{P(X_i,Y_i)}{P(X_i)P(Y_i)} \right]$$

Average of data ensemble



Probabilities for a distribution and **marginal** distributions can be calculated for a tree TN.

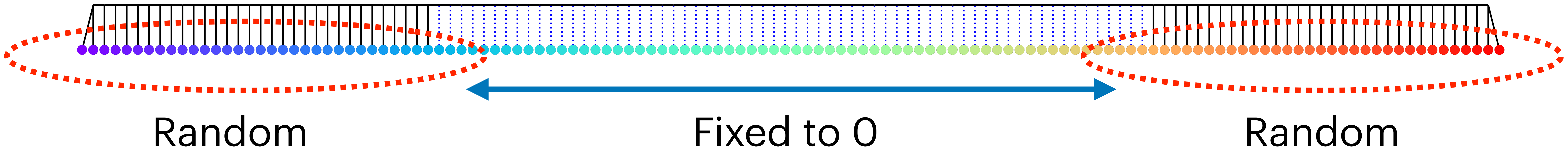
Four applications of the ATT method

- Ten random bit sequences with long-range correlations
- Images of hand-written digits (QMNIST)
- Bayesian networks
- Stock-price fluctuations in the S&P 500 index

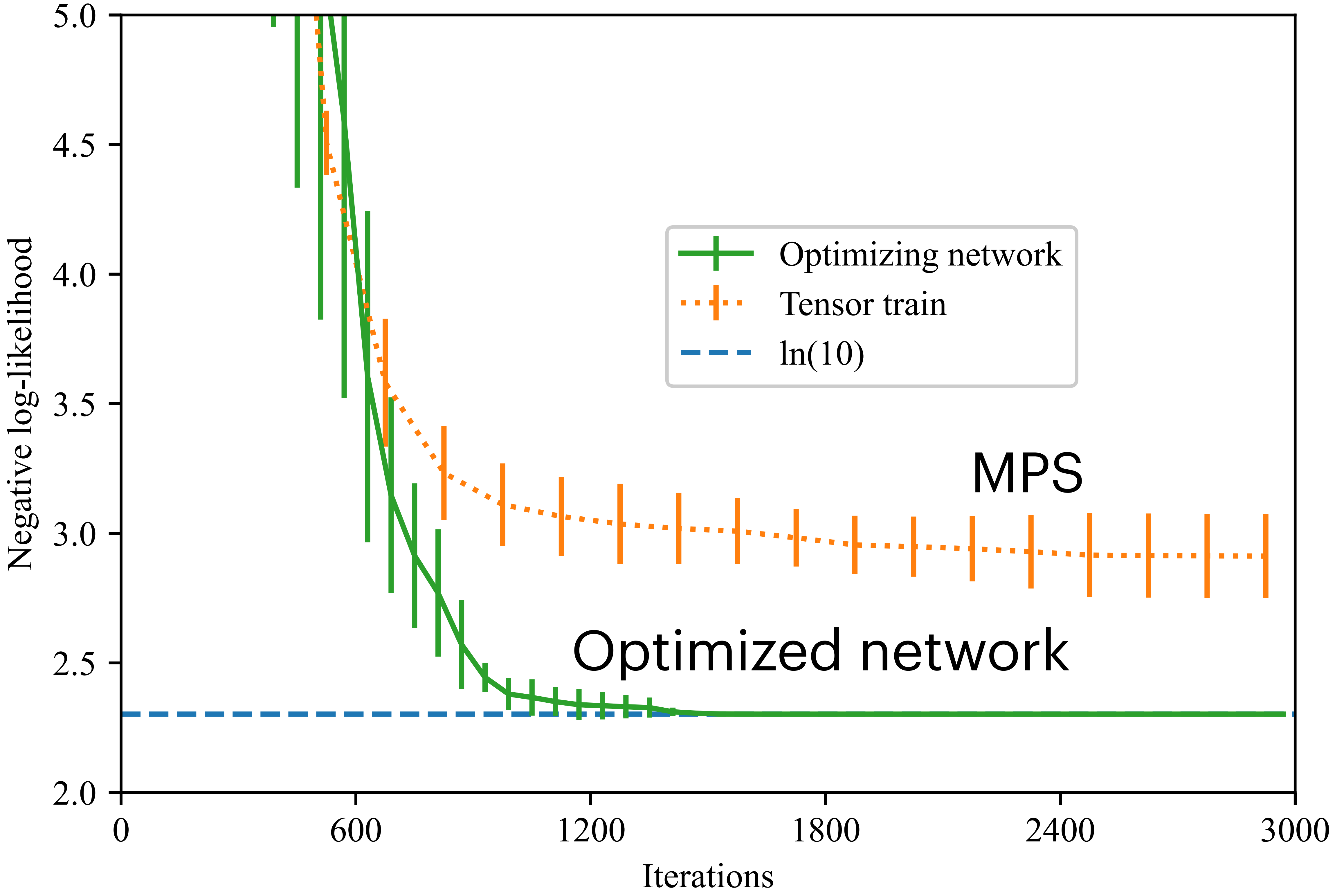
Tensor tree learns hidden relational structure in data to construct generative models

Application to random bit sequences with long-correlations

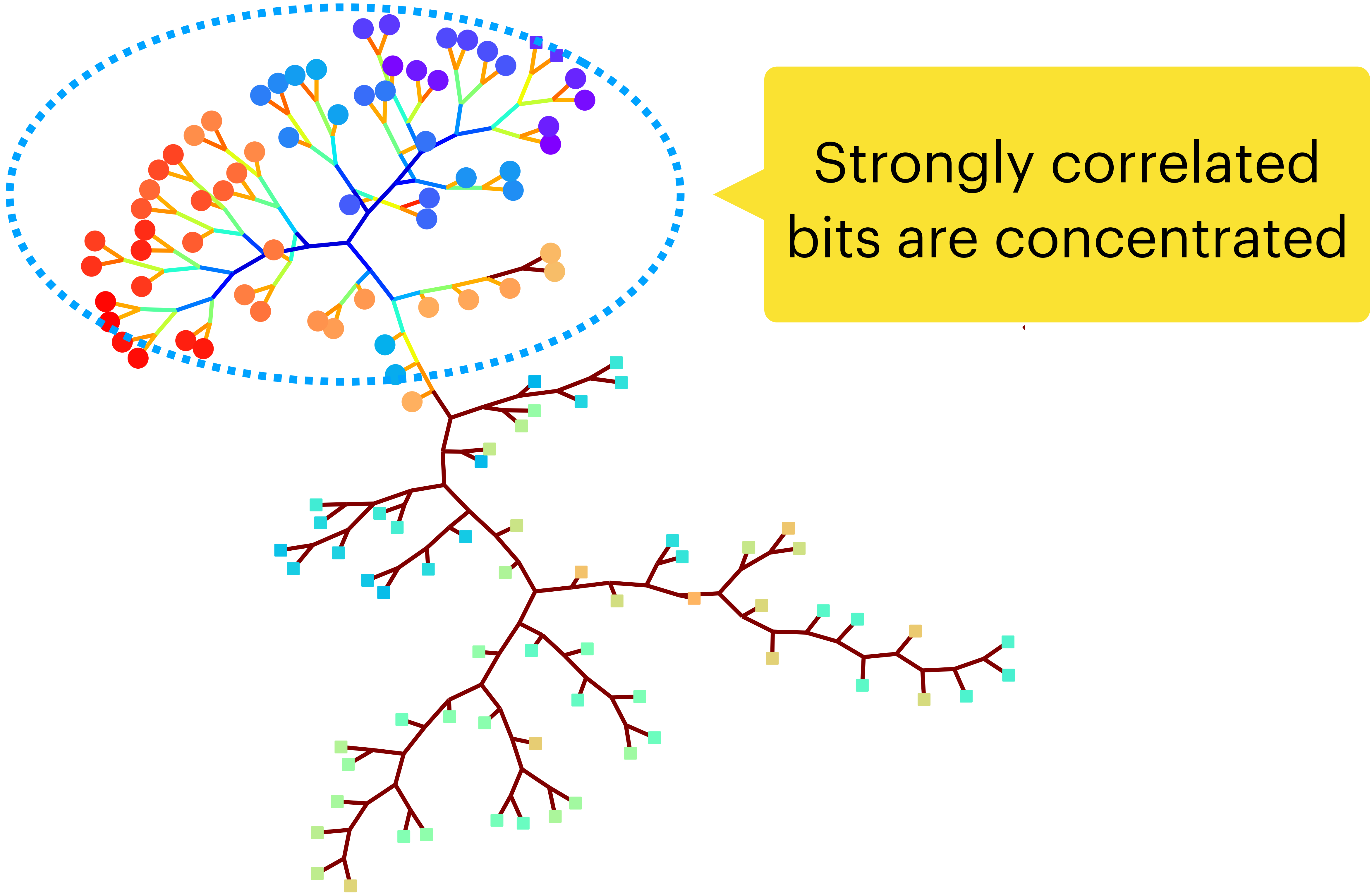
Data: ten random bit sequences



NLL in a training process

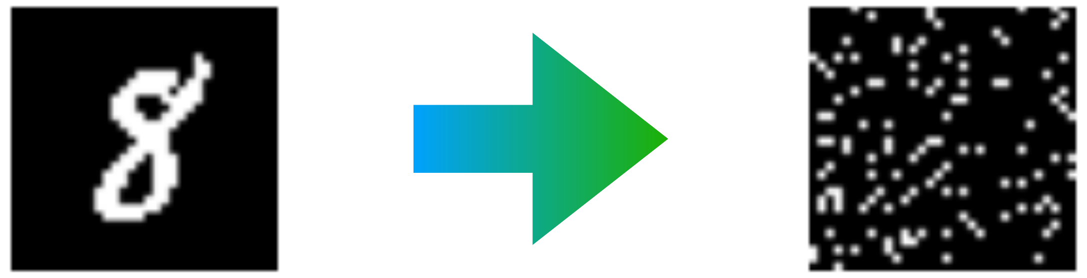


Optimized network structure

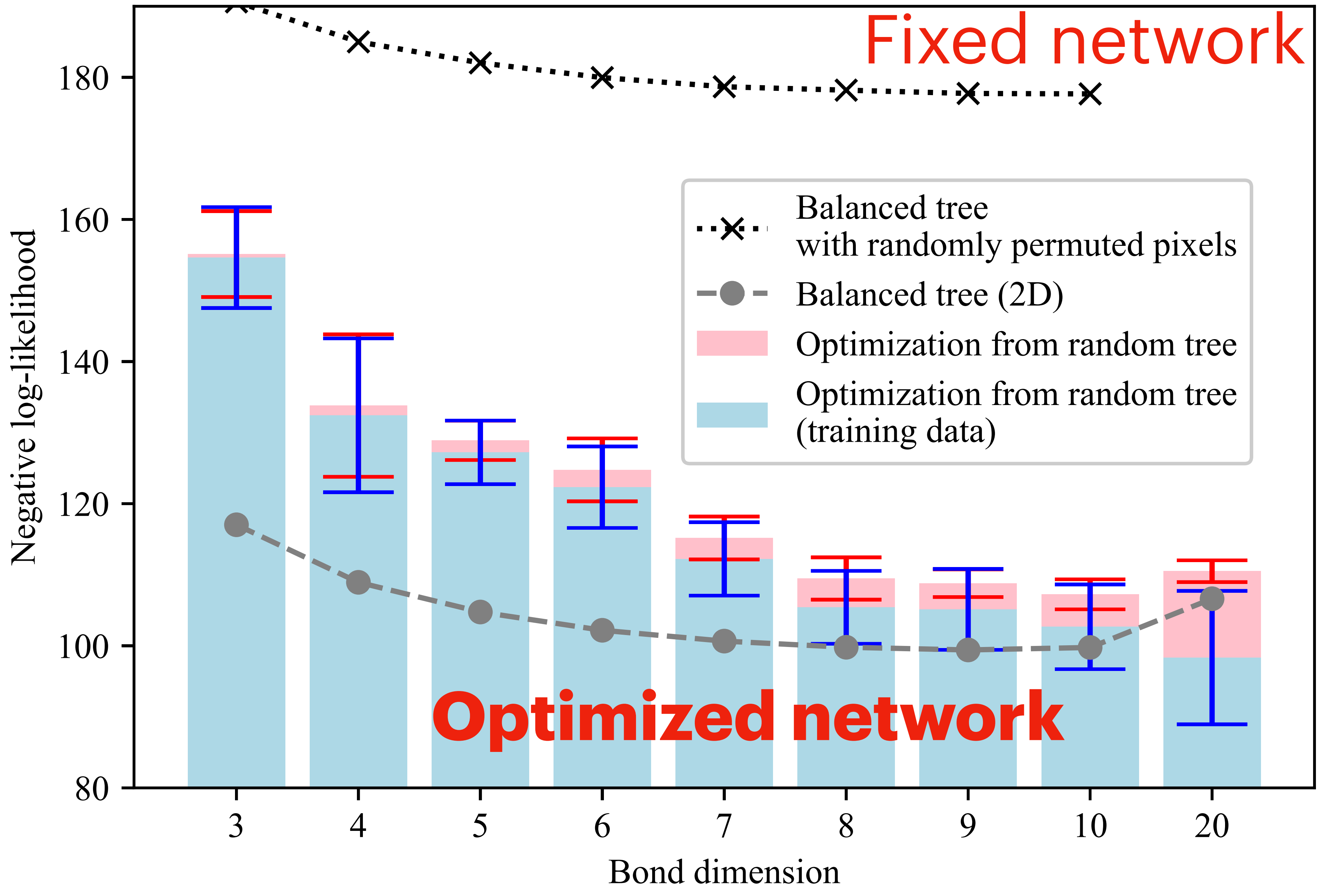


Application to images of digits

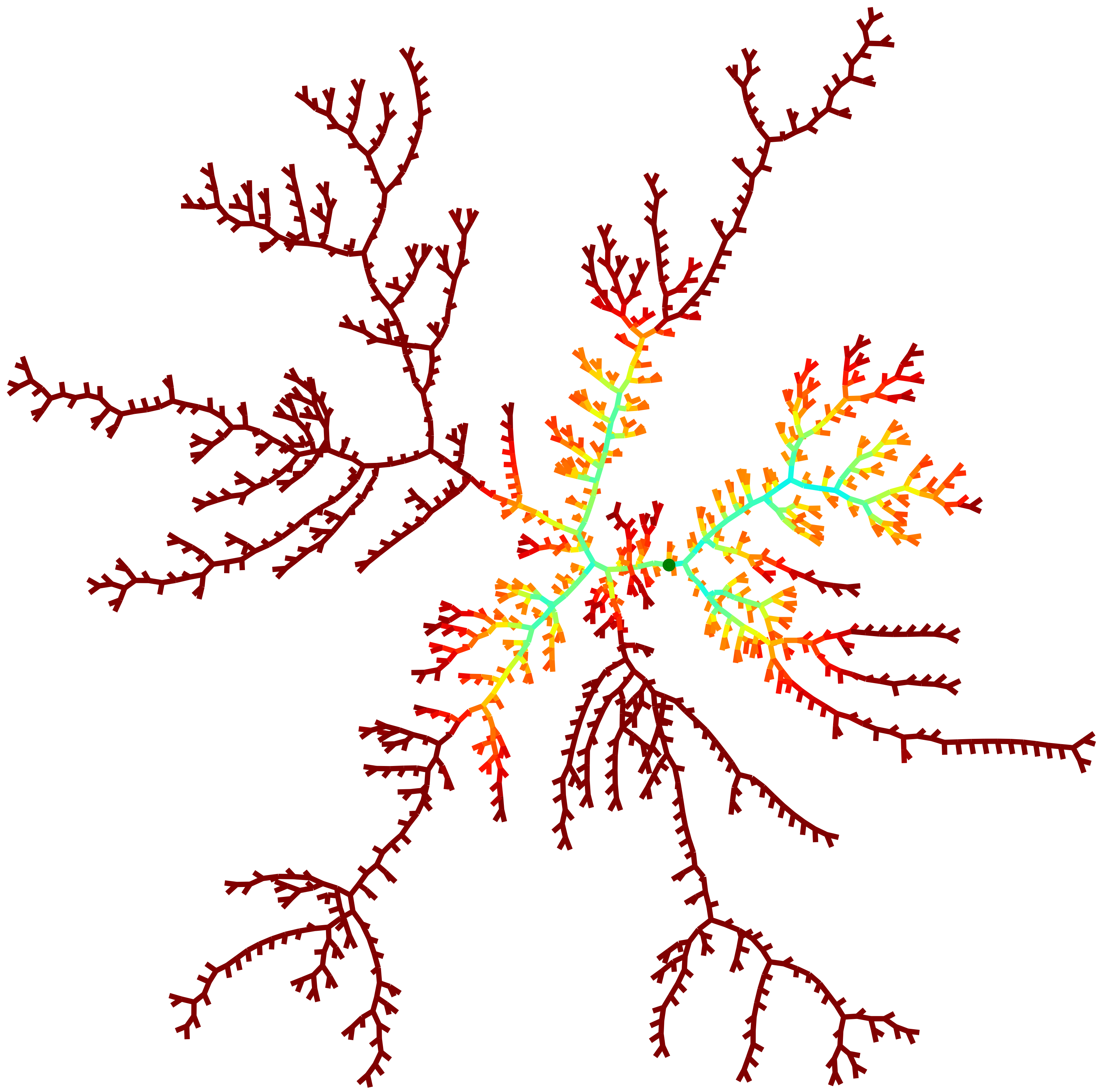
Data with a random permutation of pixels



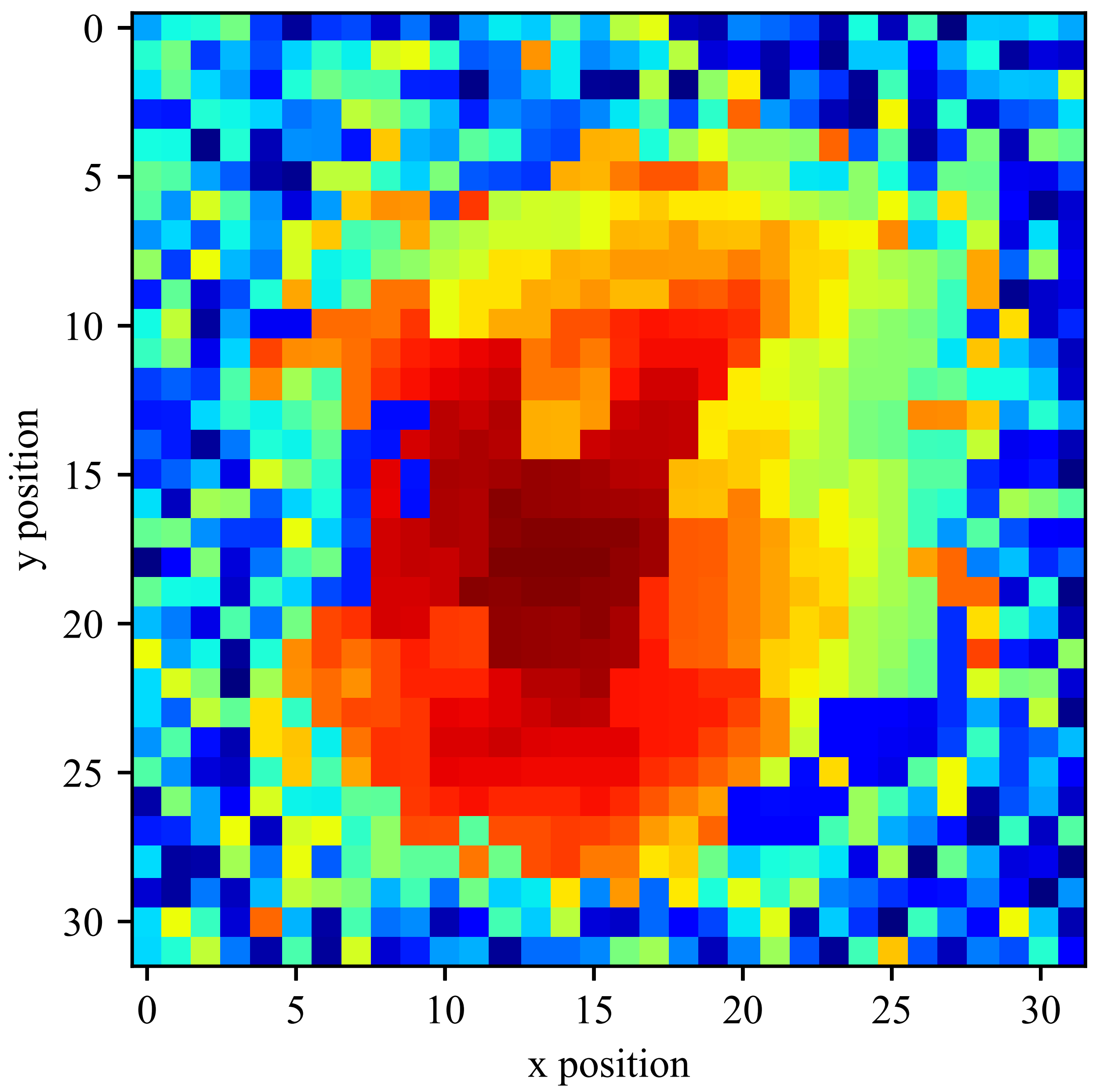
Results of NLL



Optimized network structure



Distance from the center of the network



Strongly correlated pixels are concentrated.

The ATT method automatically learns the relevant relational structure among the random variables and places them close together on the tensor network

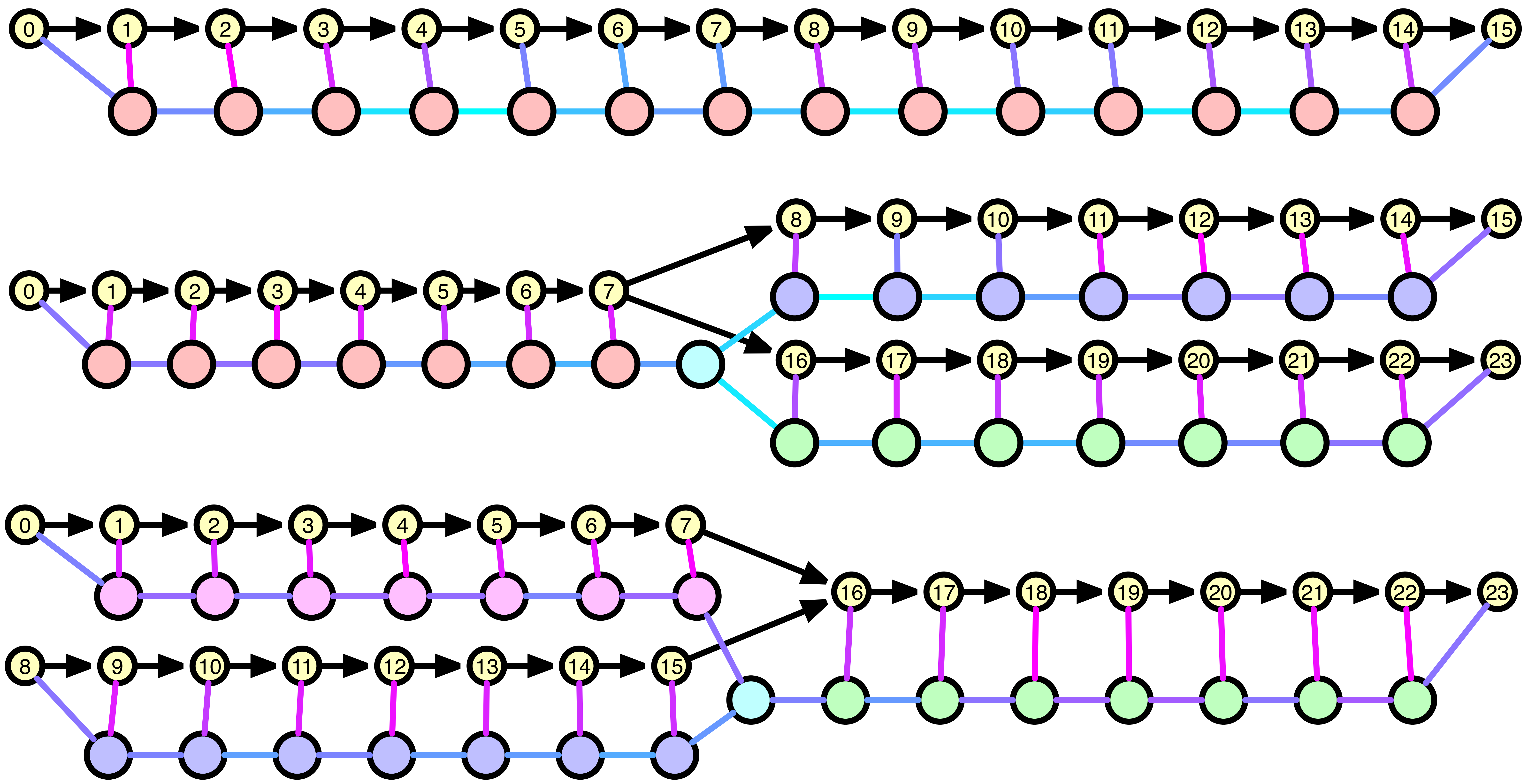
Application to Bayesian network's data

Graphical model (Bayesian network)

$$P_{\mathrm{data}}(\mathbf{x}) = \prod_i P(x_i | \{x_p\}_{p \in \mathrm{Parent}})$$

Causal dependencies among random variables

Results

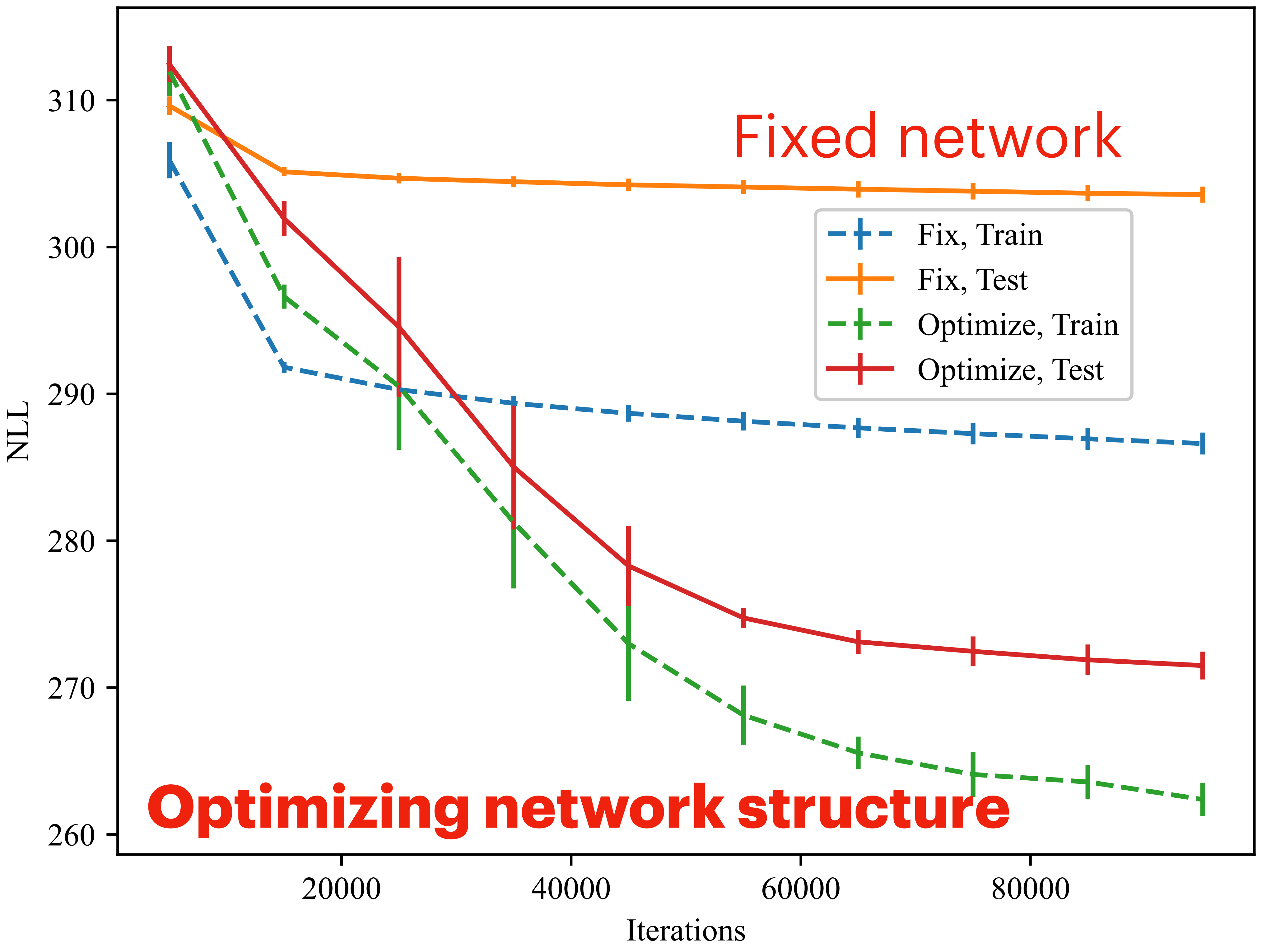


The ATT method successfully captures the corresponding topology of Bayesian networks

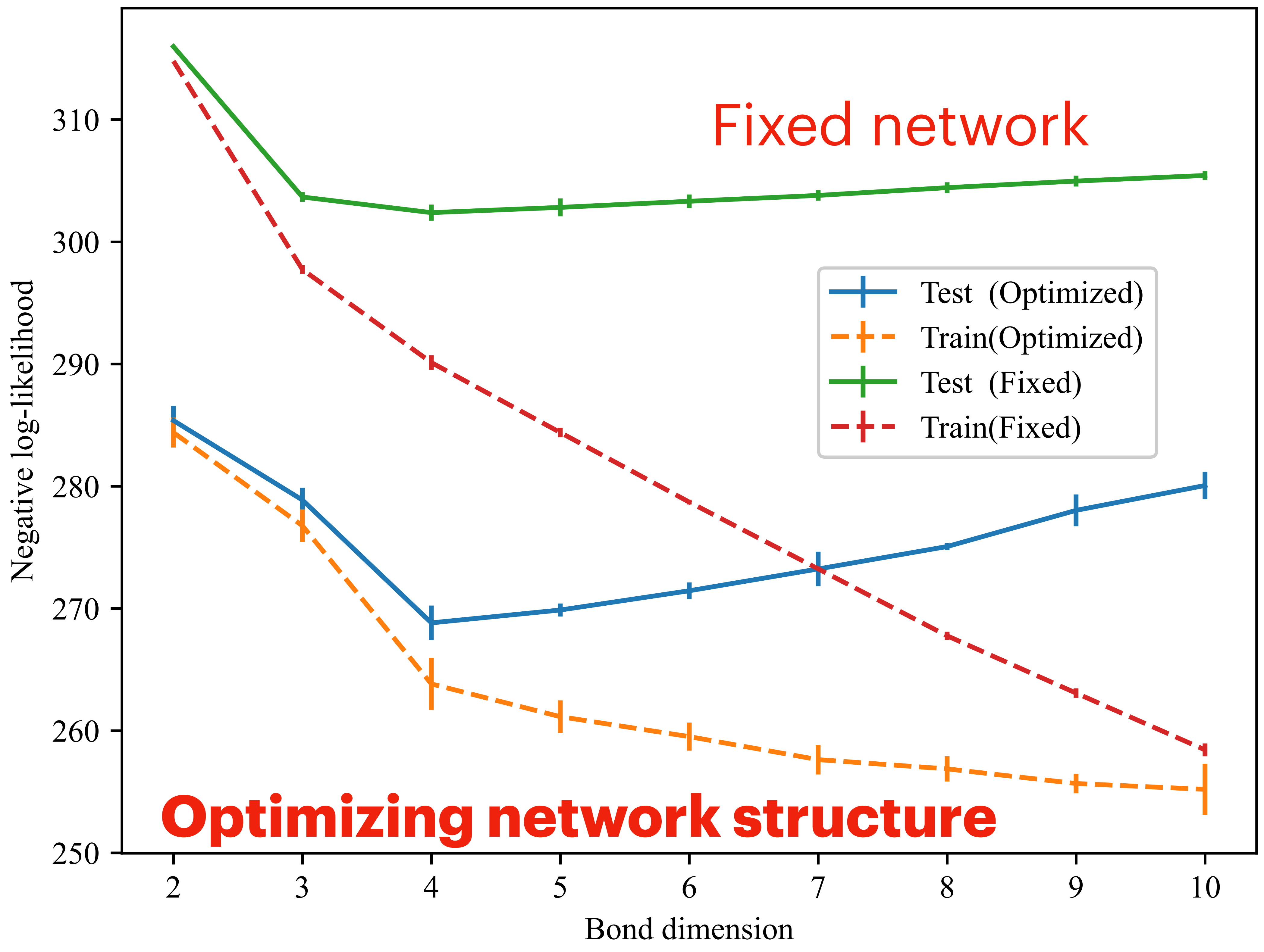
Application to stock-price fluctuation in S&P 500 index

Data: binarized change rates of stock prices 1 if it is higher than the average for all stocks and 0 otherwise.

NLL in training process



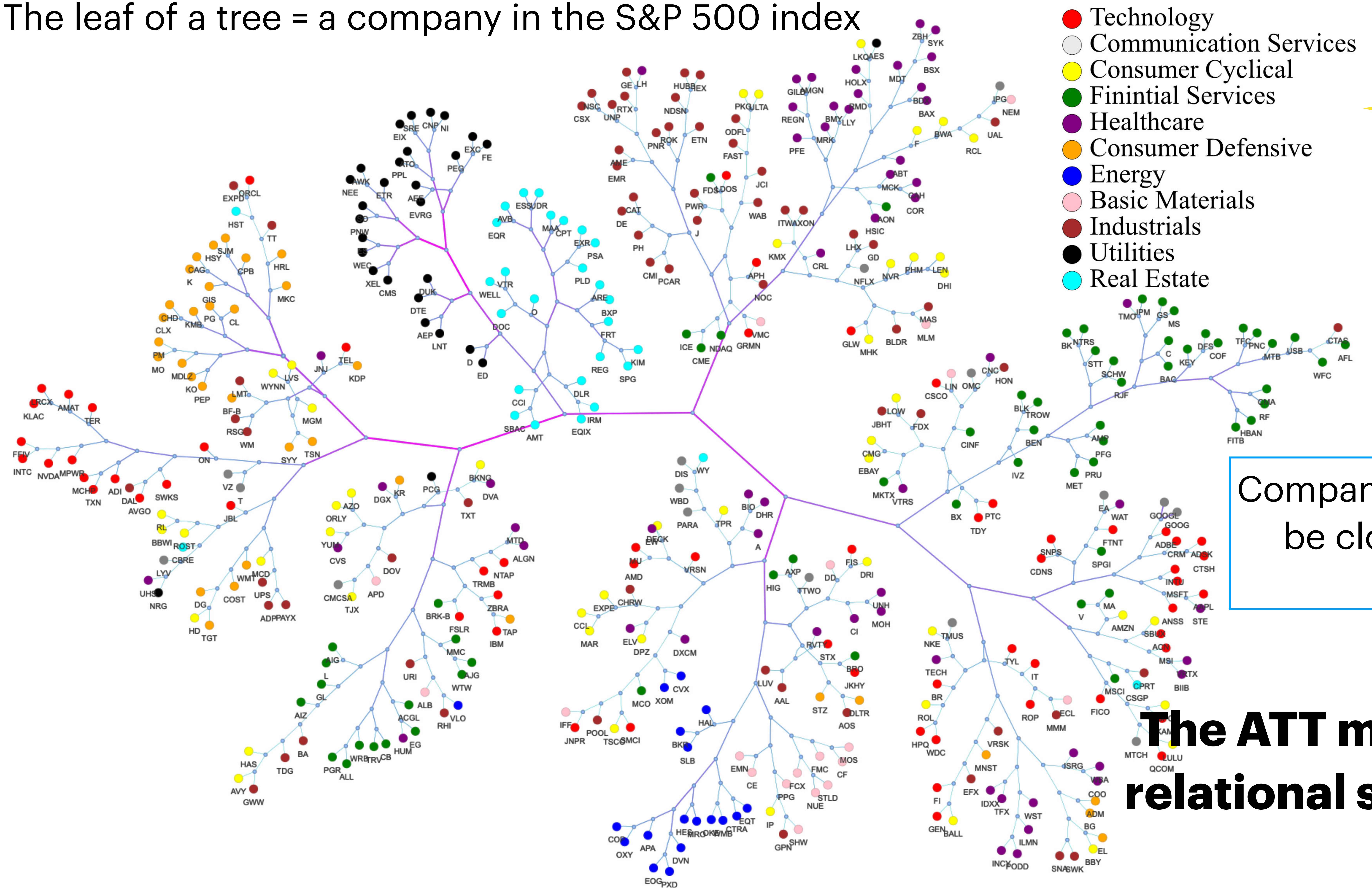
NLL vs. bond dimension



Optimized networks achieve better performance.

Optimized network structure for stock-price fluctuation in S&P500 index

The leaf of a tree = a company in the S&P 500 index



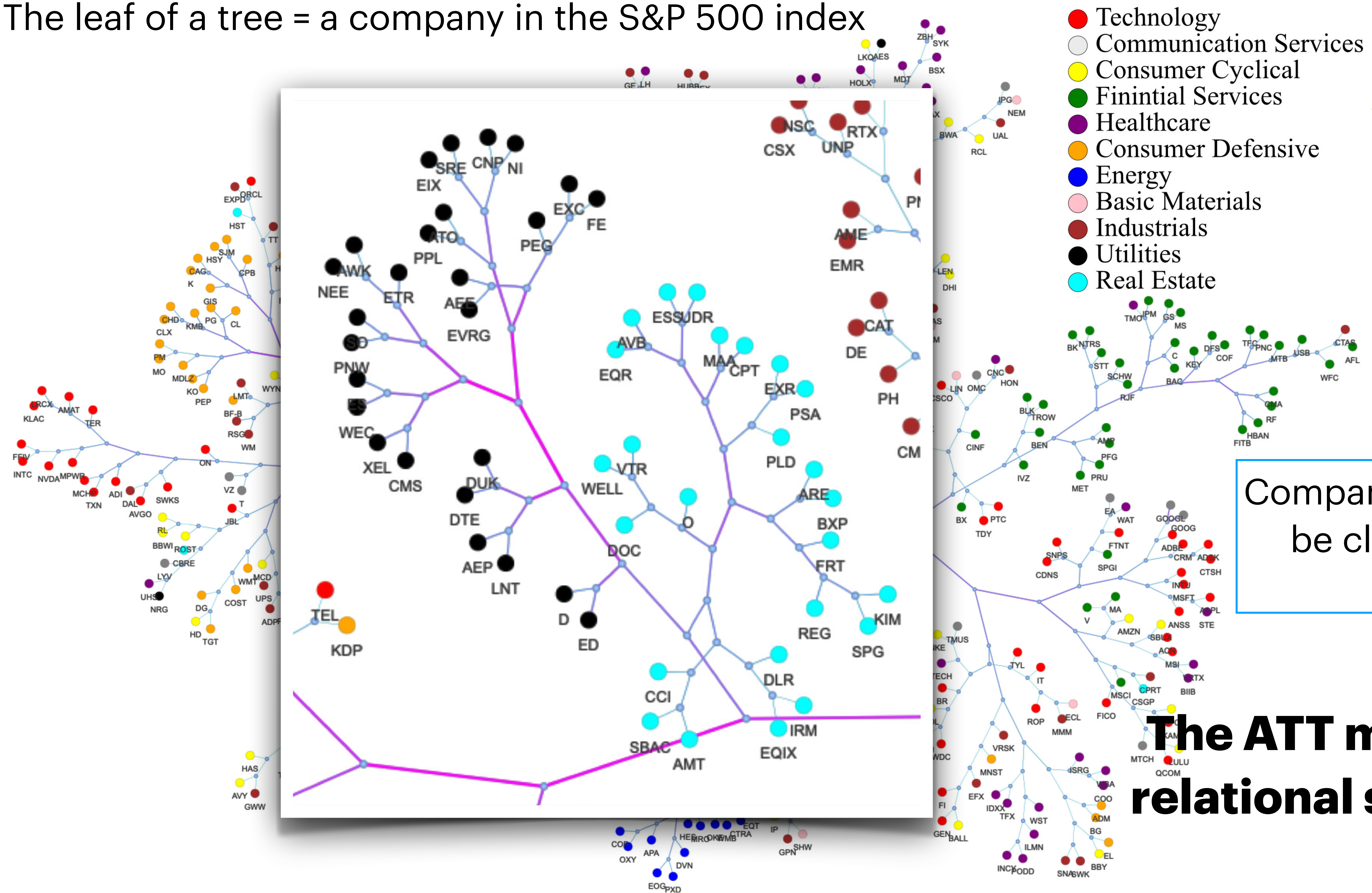
Companies are colored according to the sector to which they belong

Companies in the same sector tend to be close and form almost single-colored sub-trees.

The ATT method learns hidden relational structure behind data

Optimized network structure for stock-price fluctuation in S&P500 index

The leaf of a tree = a company in the S&P 500 index



Companies are colored according to the sector to which they belong

Companies in the same sector tend to be close and form almost single-colored sub-trees.

The ATT method learns hidden relational structure behind data

Summary

- Adapted tensor tree generative modeling
  - Succeeds for data with no prior knowledge
  - Optimized network structure shows hidden relational structure behind data

Tensor tree learns hidden relational structure in data to construct generative models

Reference: **arXiv:2408.10669**

Sample code:
https://github.com/KenjiHarada/adaptive-tensor-tree-generative-modeling