

Theory-driven quantum machine learning for HEP

Jack Y. Araz

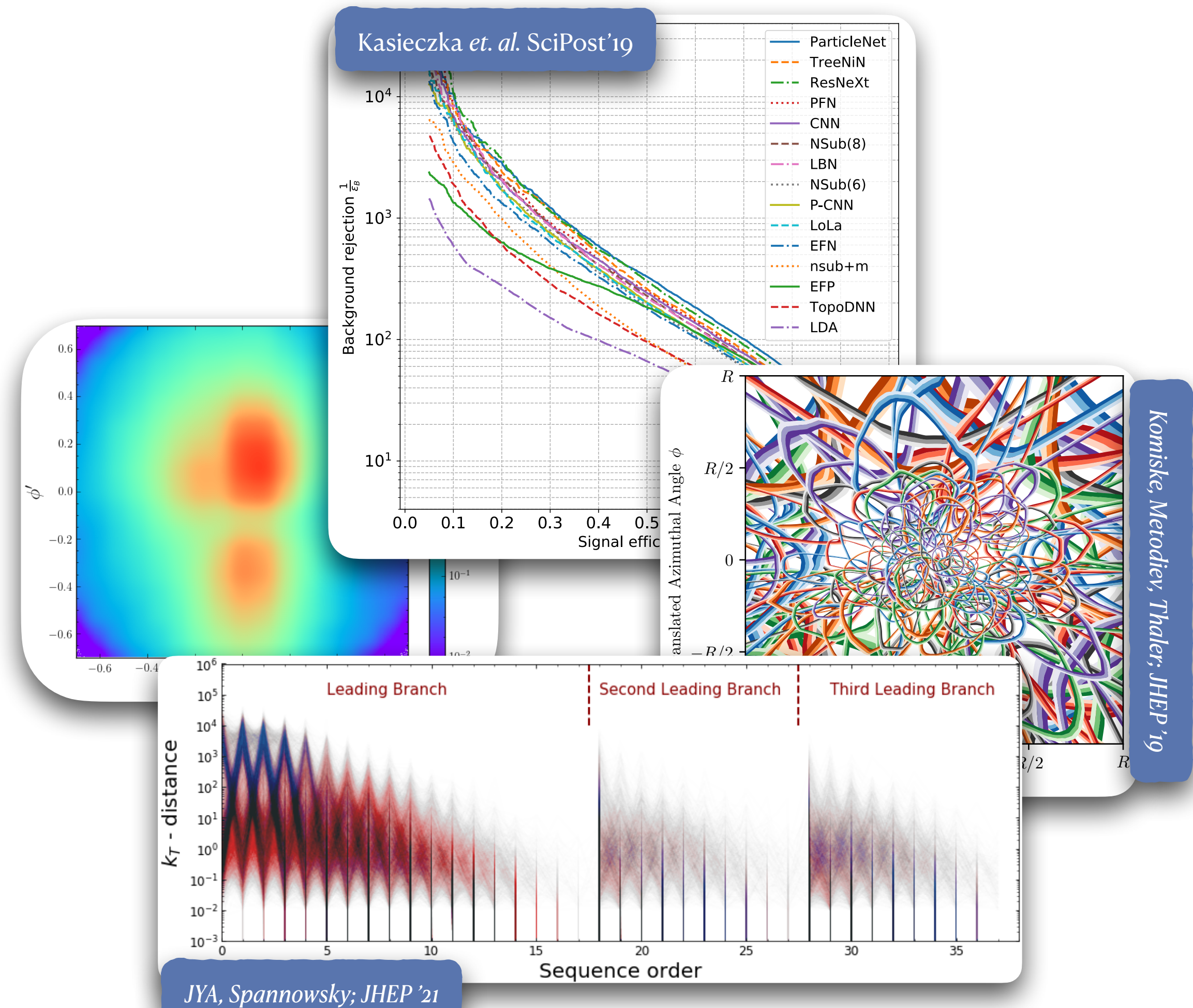
INSTITUTE FOR PARTICLE PHYSICS PHENOMENOLOGY
DURHAM UNIVERSITY

Durham University
September 19th, 2023



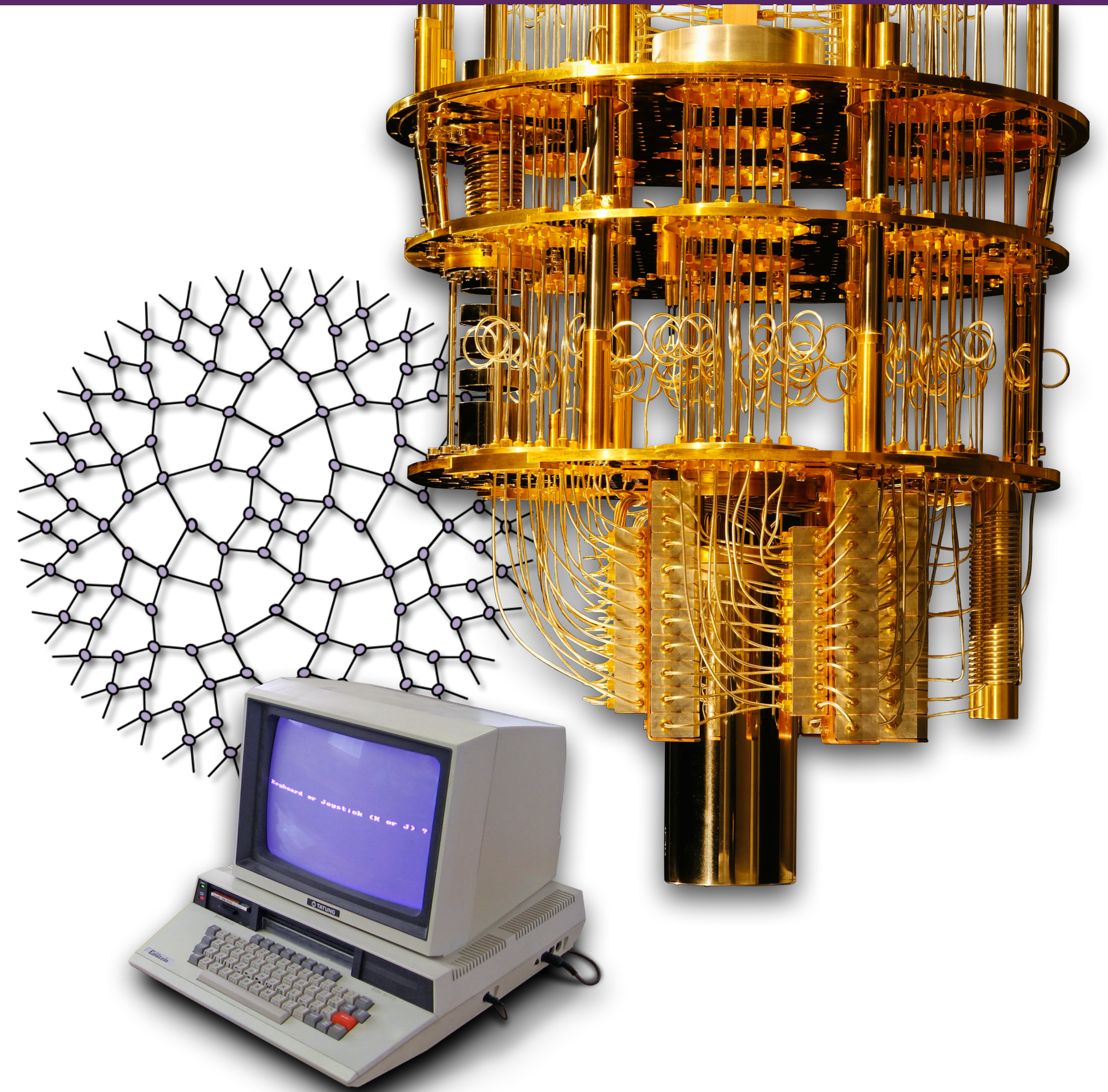
Sales pitch of the talk!

- We more or less know how to get a well-performing Neural Network to classify jets, generate LHC events, and even differentiate cats and dogs...
- What we don't know is what this network learns.
- Can we use **Quantum Mechanics** to have more insight into the learning process?
 - ◆ What has a model learned?
 - ◆ What is **learning**?
 - ◆ How do we develop “**insightful**” algorithms?
 - ◆ How to perform this on a Quantum device?
- ❖ Can an ML problem be formulated as a **quantum many-body system**?

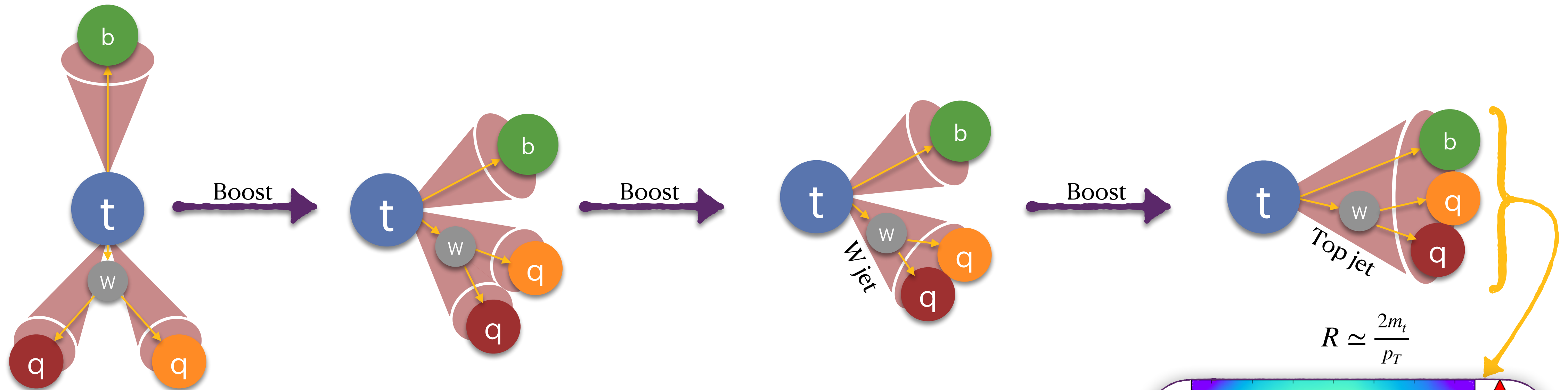


Outline

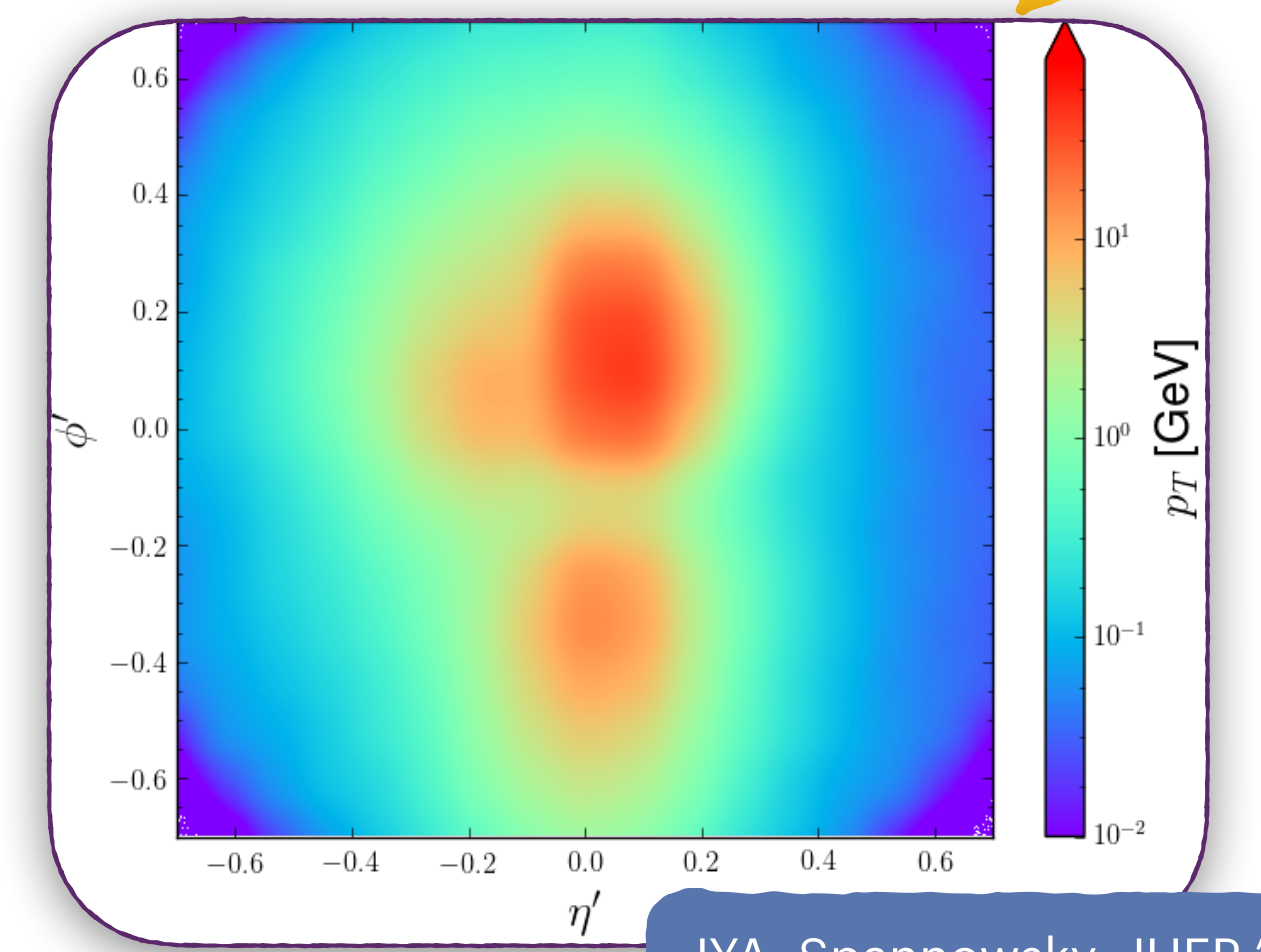
- ❖ Hamiltonian learning for anomaly detection
- ❖ Classification as a quantum many-body problem
- ❖ Conclusion



Hello world of HEP-ML: Top tagging



- ❖ With the increased boost factor, jets (top decay products) are getting more collimated.
- ❖ Hadronic top tagging tools: Mass grooming and filtering, Pruning, Trimming, Soft Drop Tagger, Mass Drop Tagger, HEPTopTagger, Machine Learning



JYA, Spannowsky; JHEP '21

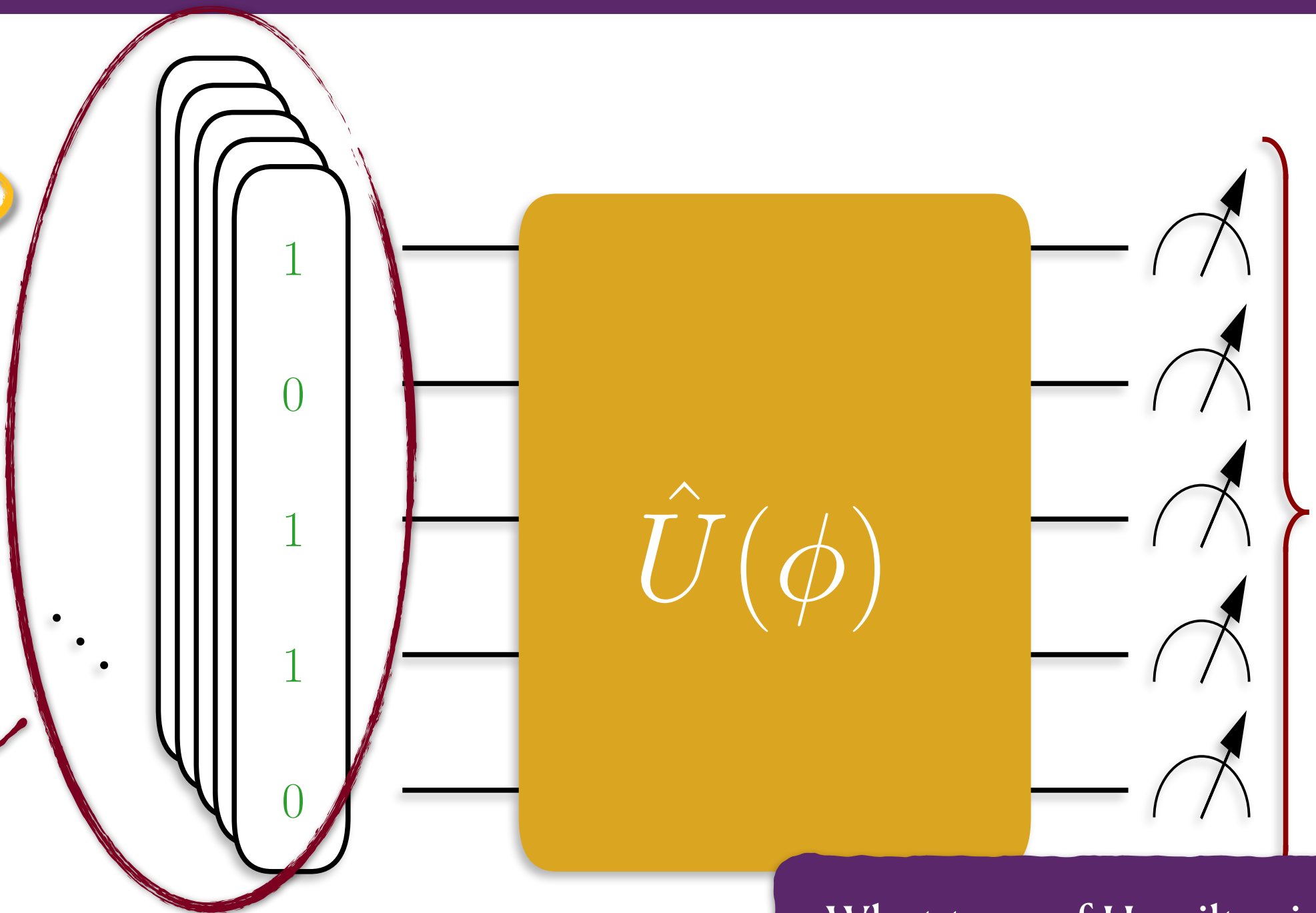
Hamiltonian learning for anomaly detection

What has Hamiltonian to do with data?

JYA, Spannowsky; arXiv: 2211.03803

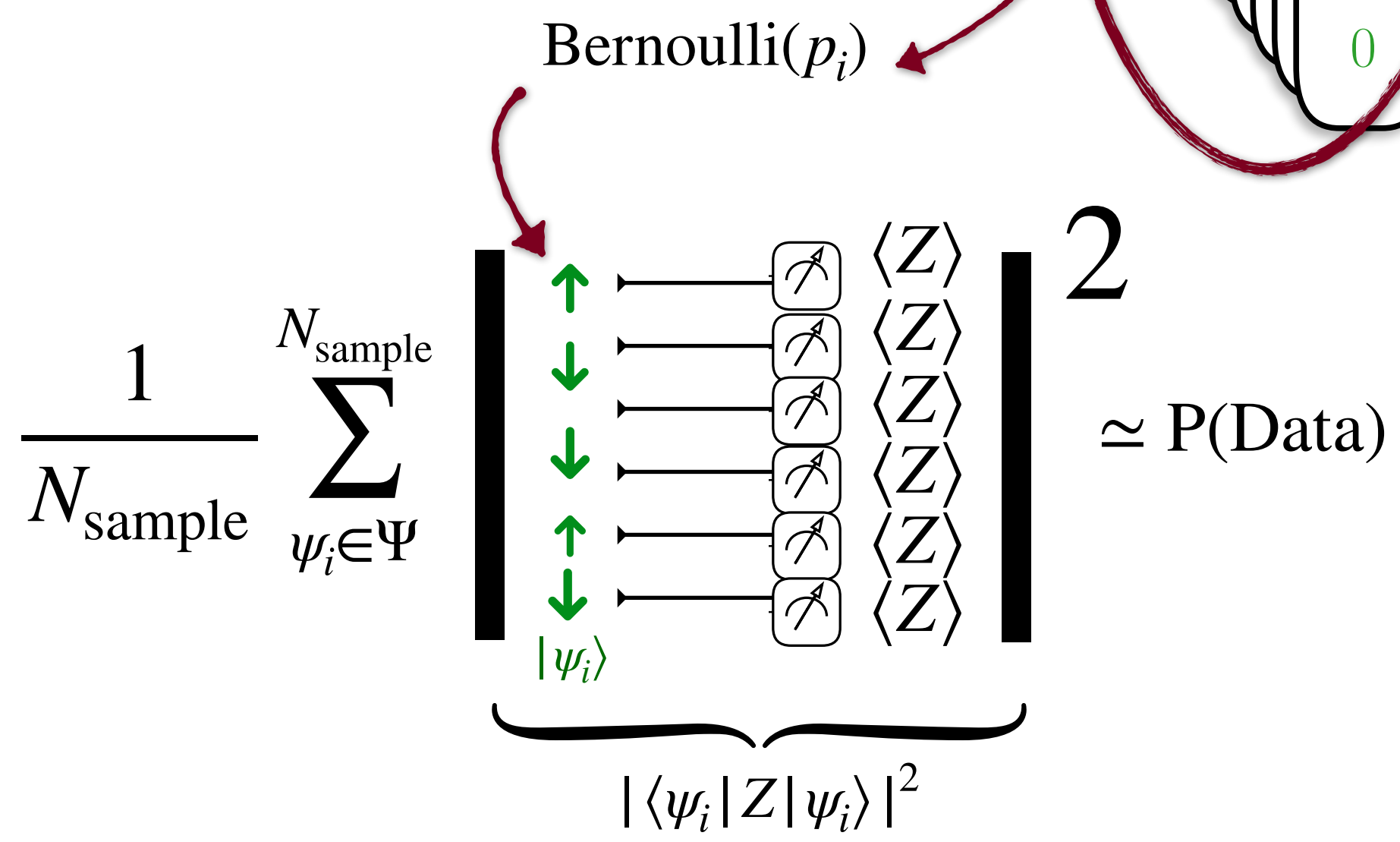
Quantum Circuit is a pure-state simulator!

A data point can be represented as a mixed state
 $\sigma_D = \sum_i p_i |\psi_i\rangle$, $|\psi_i\rangle :=$ pure states



$$\langle \hat{K} \rangle_{\theta, \phi}$$

Hamiltonian captures the entropic probability distribution of the mixed state.



What type of Hamiltonian can we choose?

- ❖ Any field theory Hamiltonian, e.g. Ising model
- ❖ A generic Hamiltonian e.g. $\sum_{\langle i,j \rangle} (\alpha_{i,j} \sigma_i^+ \sigma_j^- + \text{h.c.})$
- ❖ But can these options capture the full complexity of the data? Can we get ambitious?

What has Hamiltonian to do with data?

JYA, Spannowsky; arXiv: 2211.03803

Quantum Circuit is a pure-state simulator!

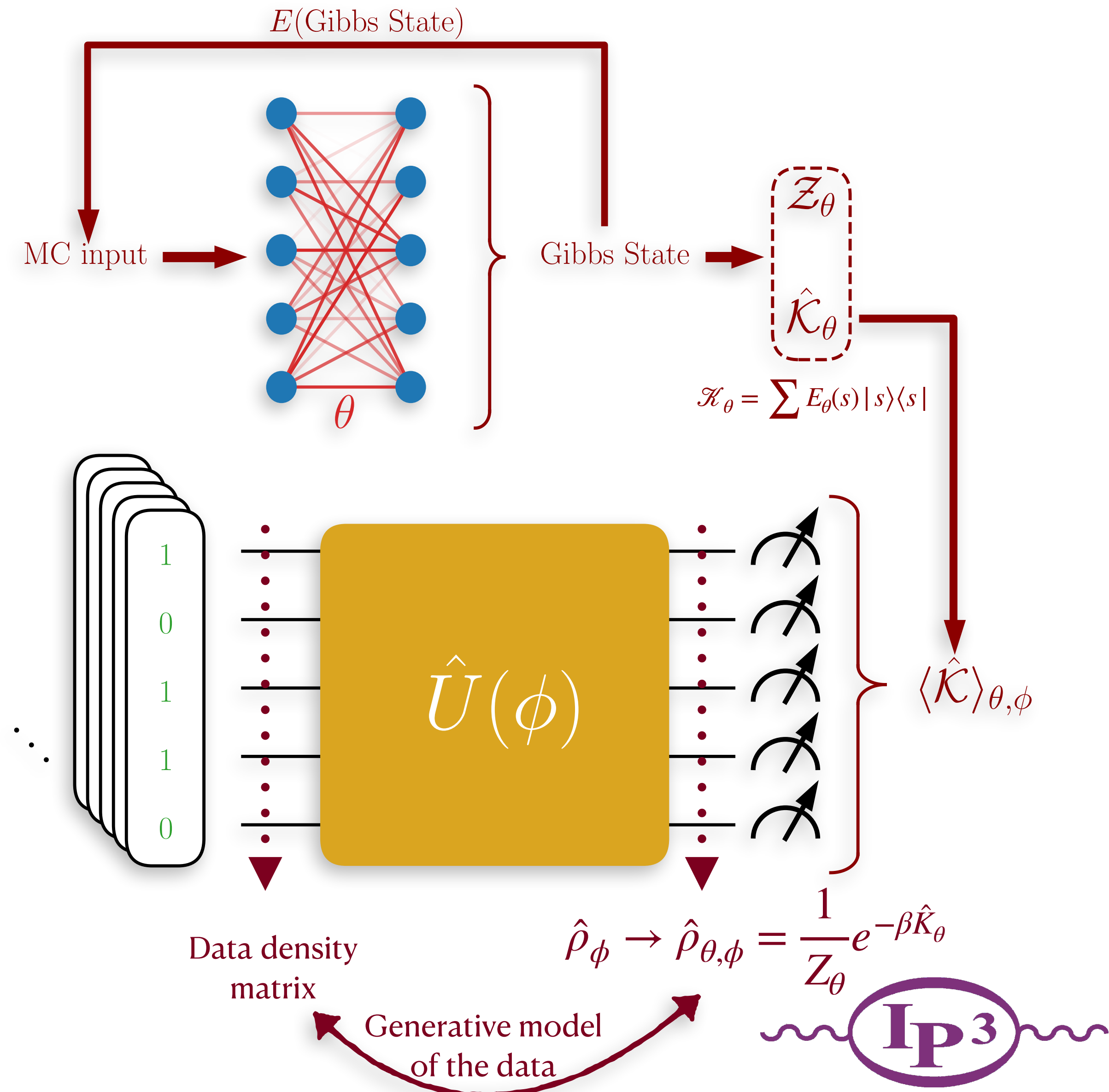
A data point can be represented as a mixed state
 $\sigma_D = \sum_i p_i |\psi_i\rangle$, $|\psi_i\rangle :=$ pure states

See Gibbs-Delbrück-Moliève variational principle
 $F = E - TS = -k_\beta T \ln Z_\theta$

$$\mathcal{L}_{\theta,\phi}(\sigma_D) = \beta \langle \hat{K} \rangle_{\theta,\phi} + k_\beta \ln Z_\theta \geq S(\sigma_D)$$

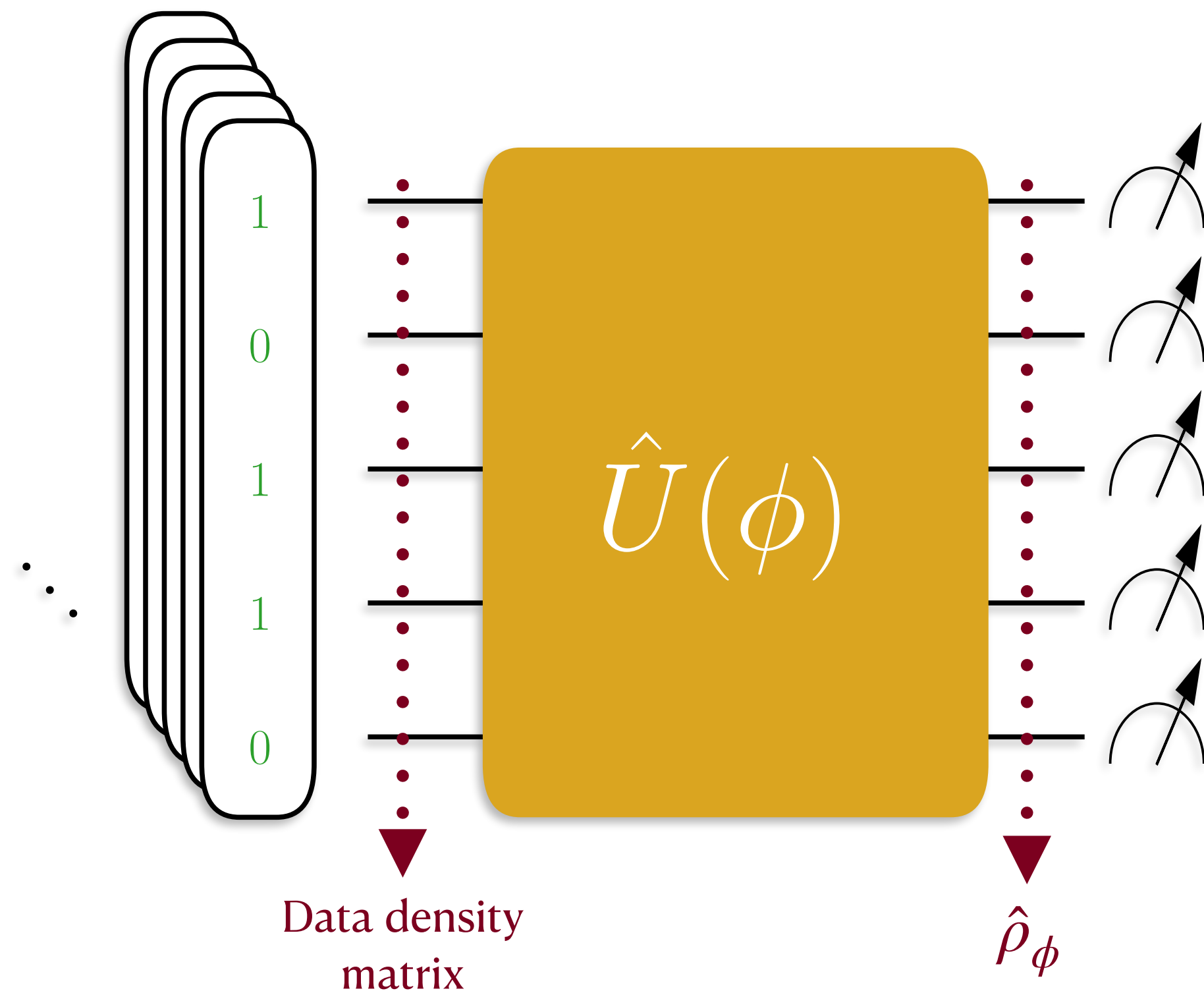
$k_\beta :=$ Boltzmann constant

$\beta :=$ Inverse temperature



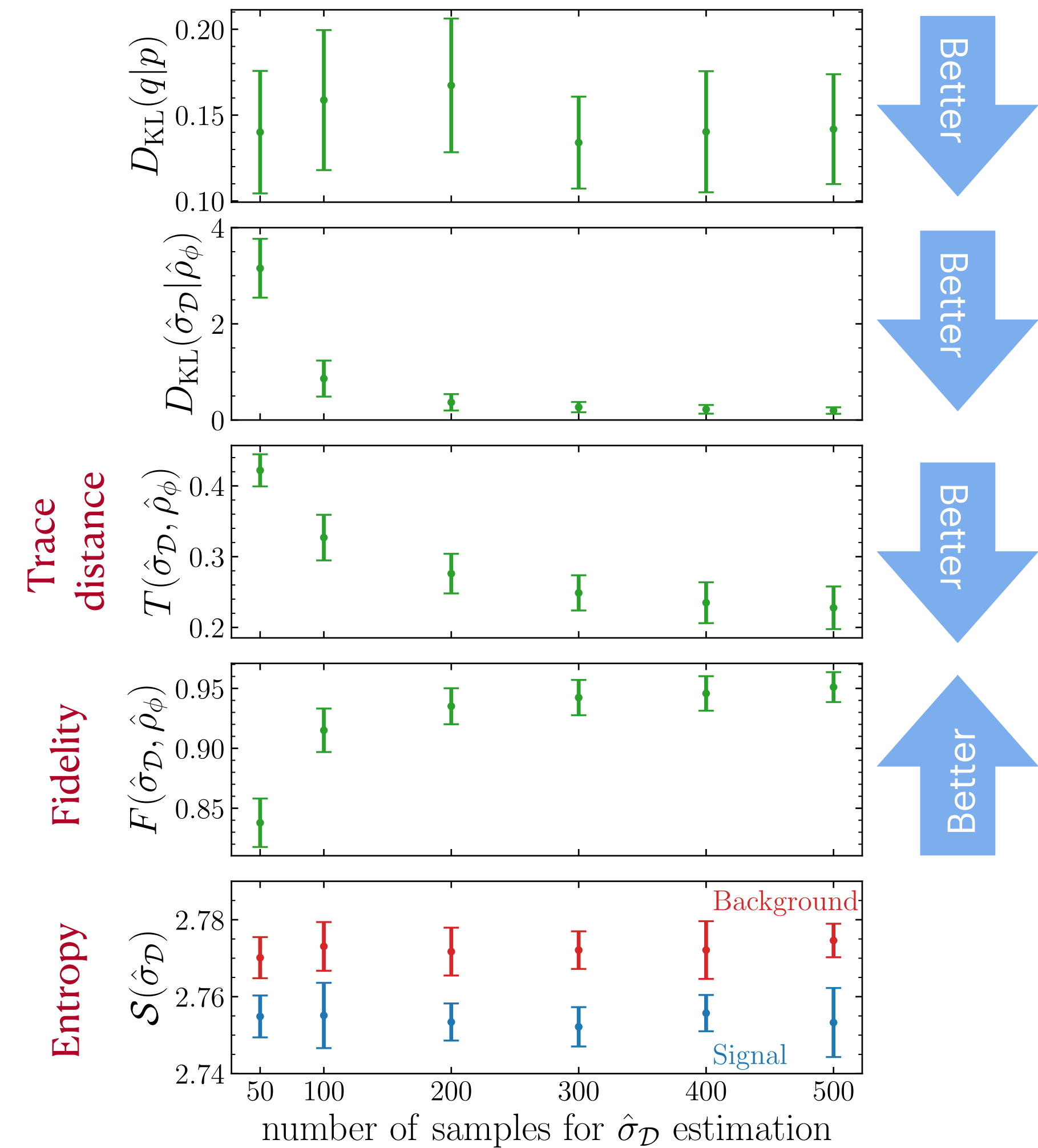
What has Hamiltonian to do with data?

JYA, Spannowsky; arXiv: 2211.03803



$$\mathcal{L}_{\theta, \phi}(\sigma_D) = \langle \hat{K} \rangle_{\theta, \phi} + \ln Z_{\theta} \geq S(\sigma_D)$$

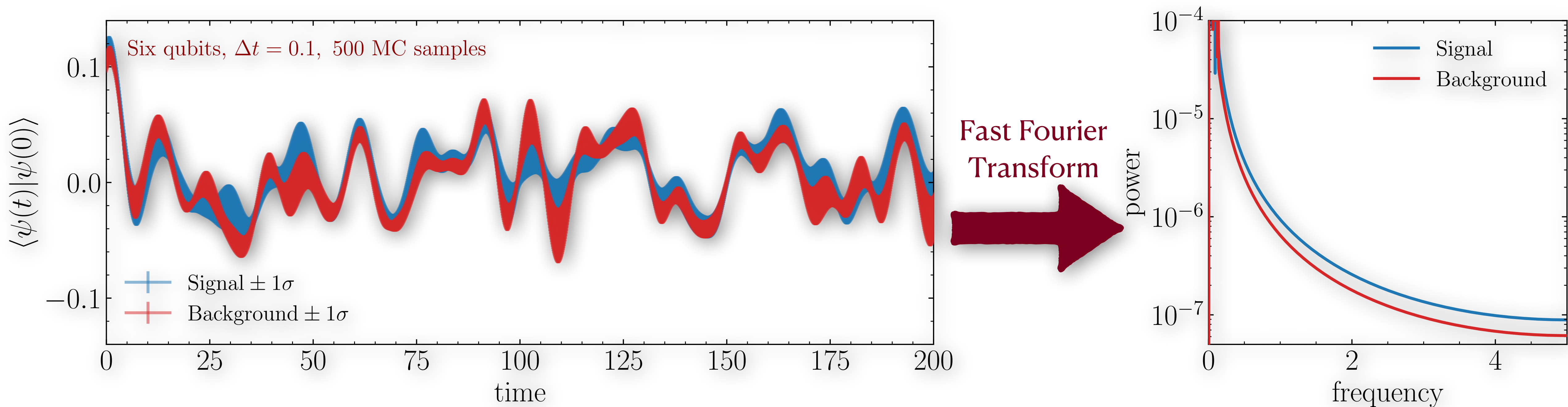
$$\arg \min_{\theta, \phi} \mathcal{L}_{\theta, \phi}(\sigma_D) \simeq S(\sigma_D)$$



Hamiltonian as a discriminator!

JYA, Spannowsky; arXiv: 2211.03803

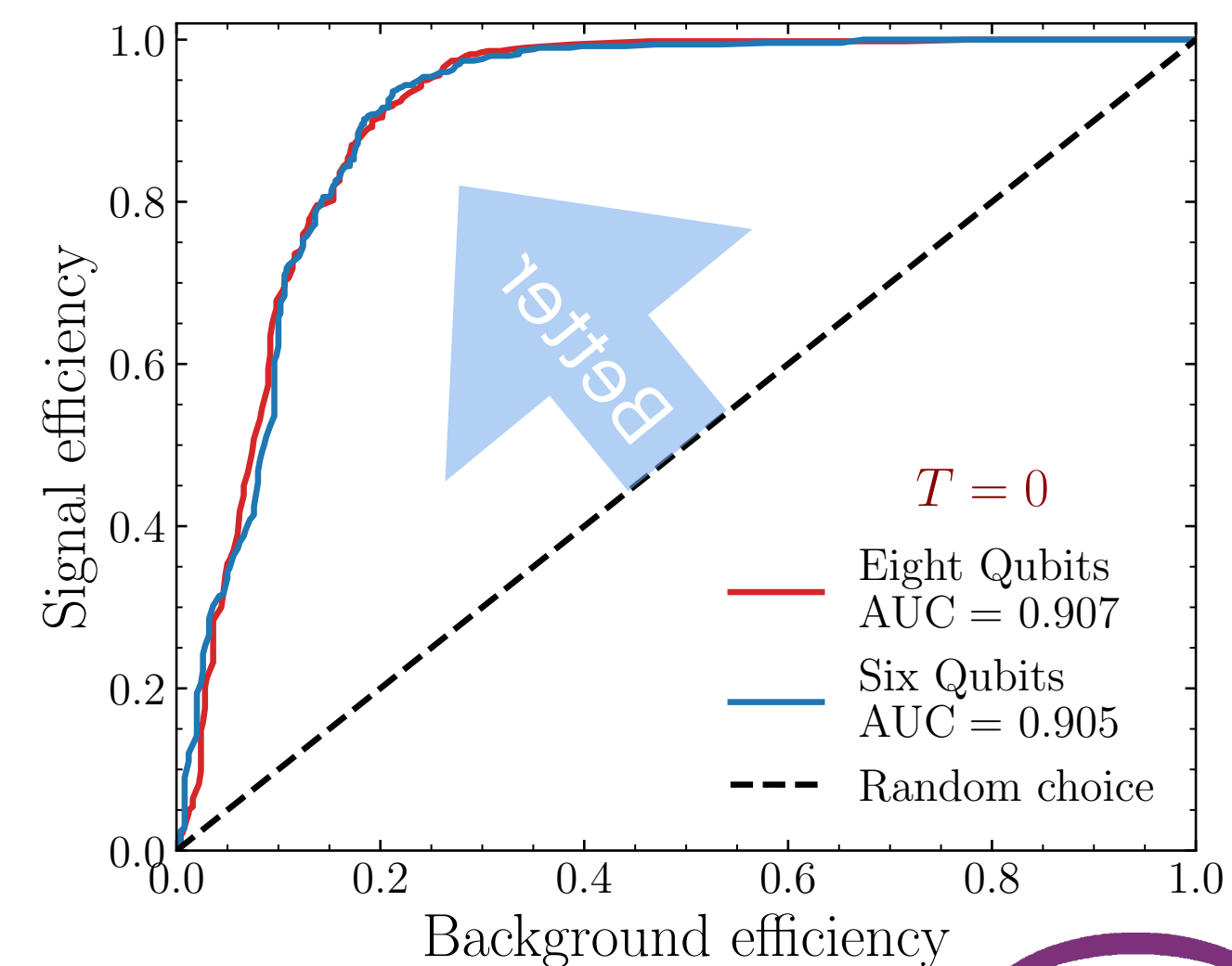
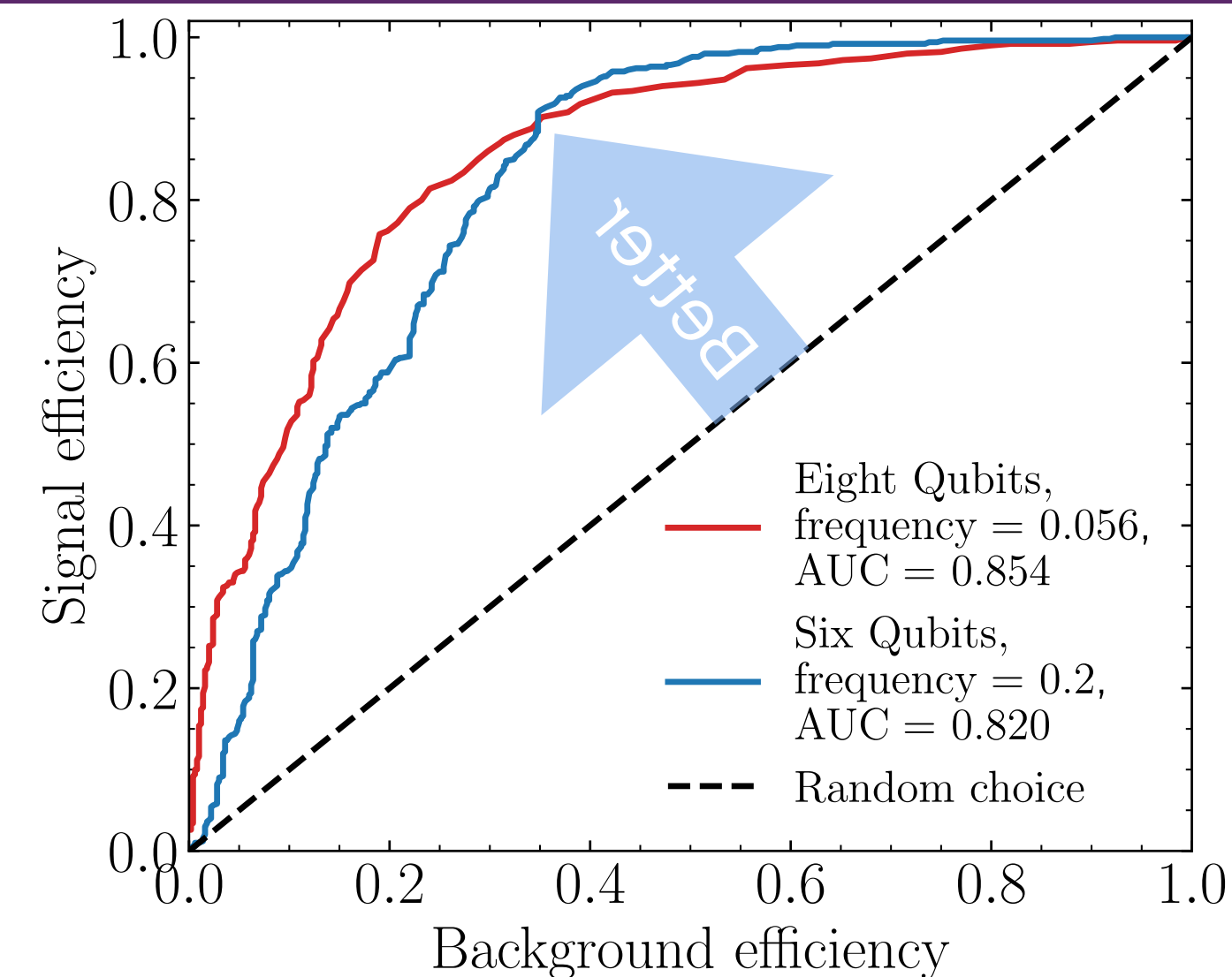
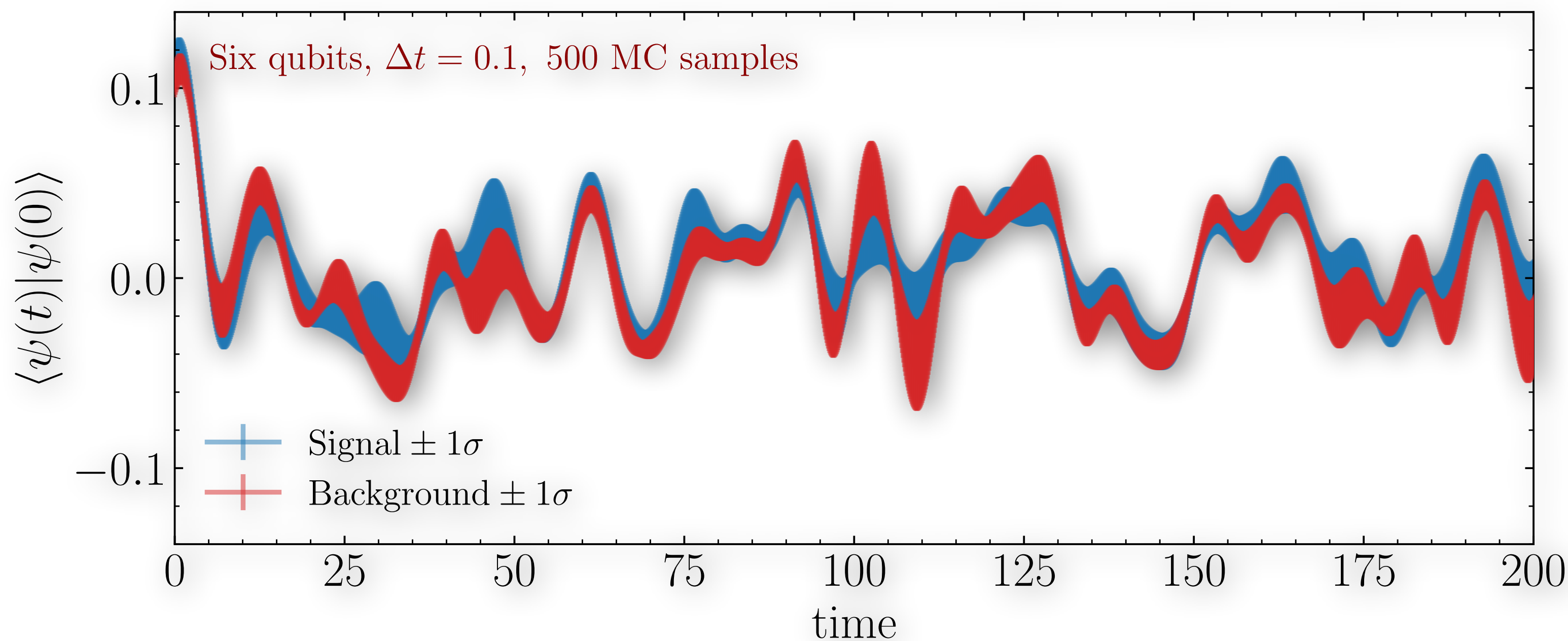
Trotter-Suzuki approximation $\longrightarrow e^{-iT\hat{K}_\theta} = \prod^N e^{-i\Delta t\hat{K}_\theta}$



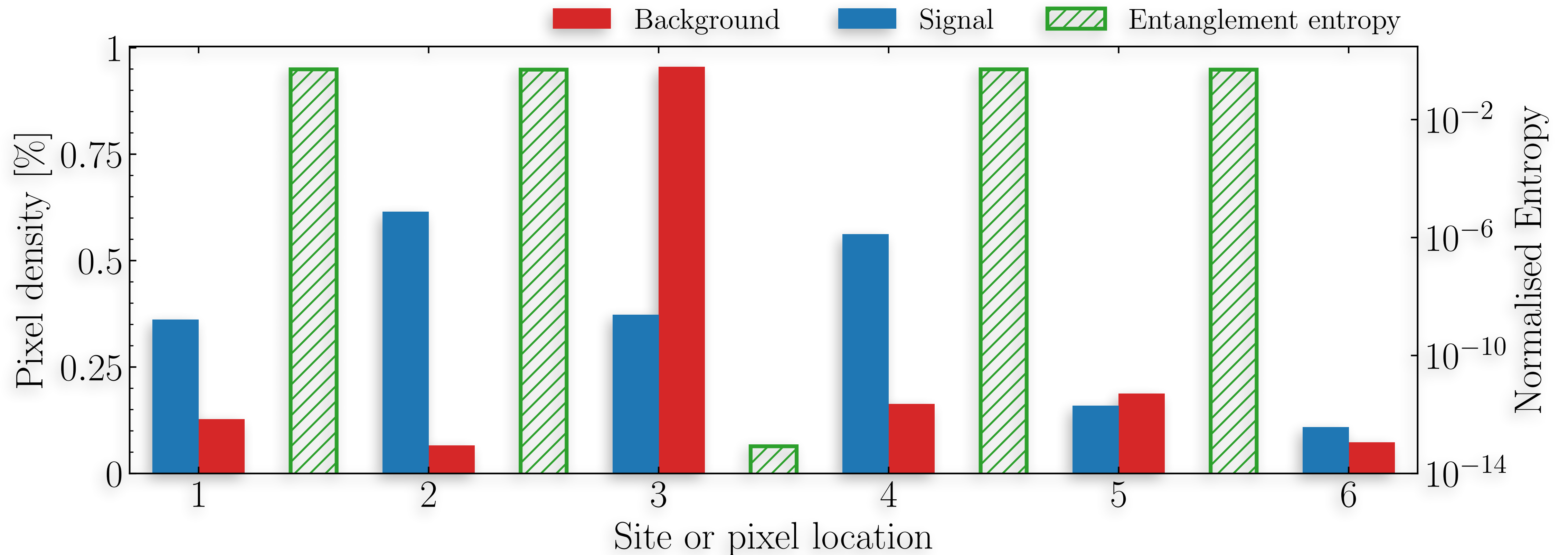
Hamiltonian as a discriminator!

JYA, Spannowsky; arXiv: 2211.03803

Trotter-Suzuki approximation $\longrightarrow e^{-iT\hat{K}_\theta} = \prod^N e^{-i\Delta t\hat{K}_\theta}$



What did the Hamiltonian learn?

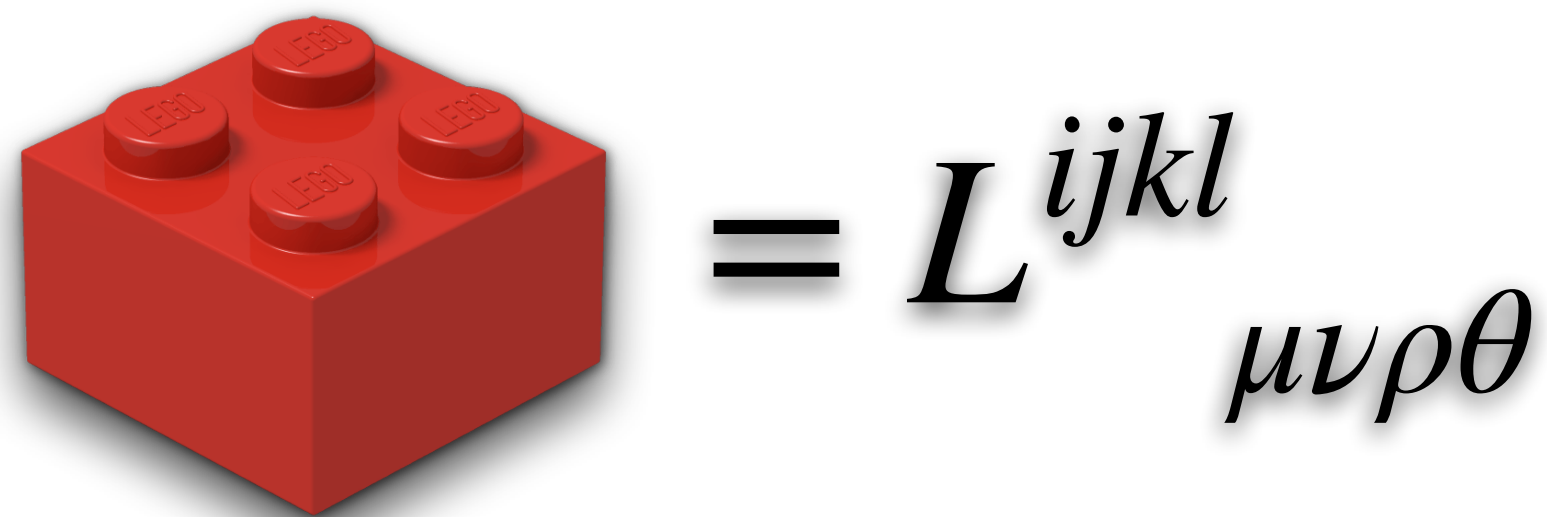
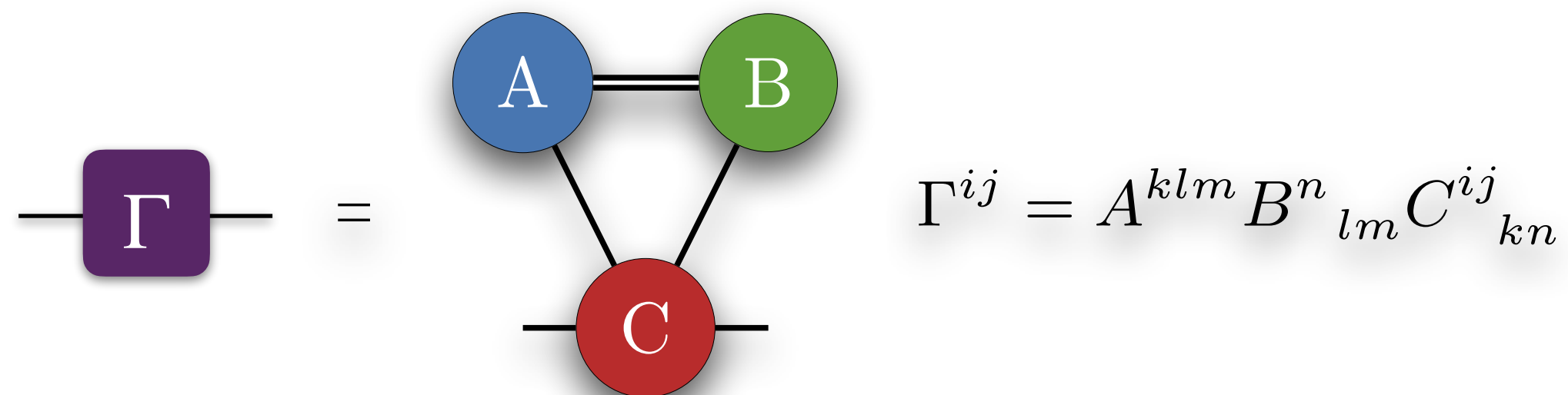
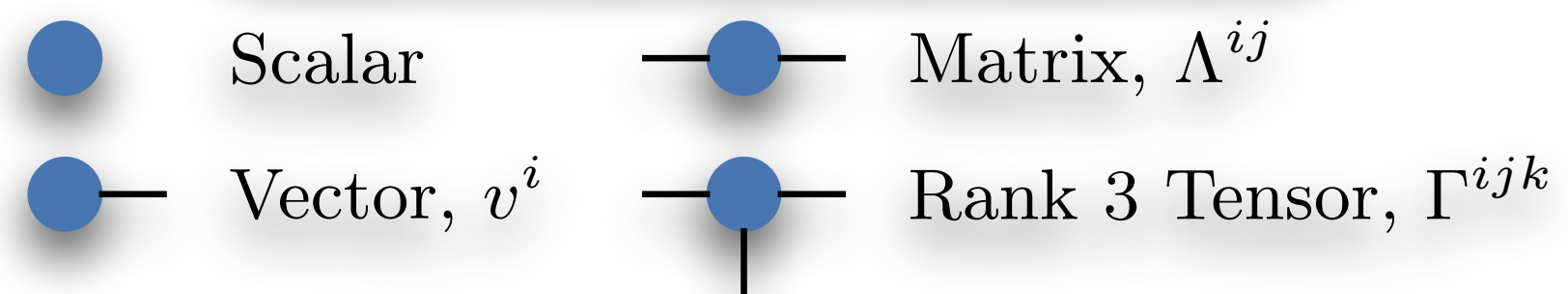


$$\mathcal{S}(\rho) = -\text{Tr}[\rho \log \rho]$$

Classification as a quantum many-body problem

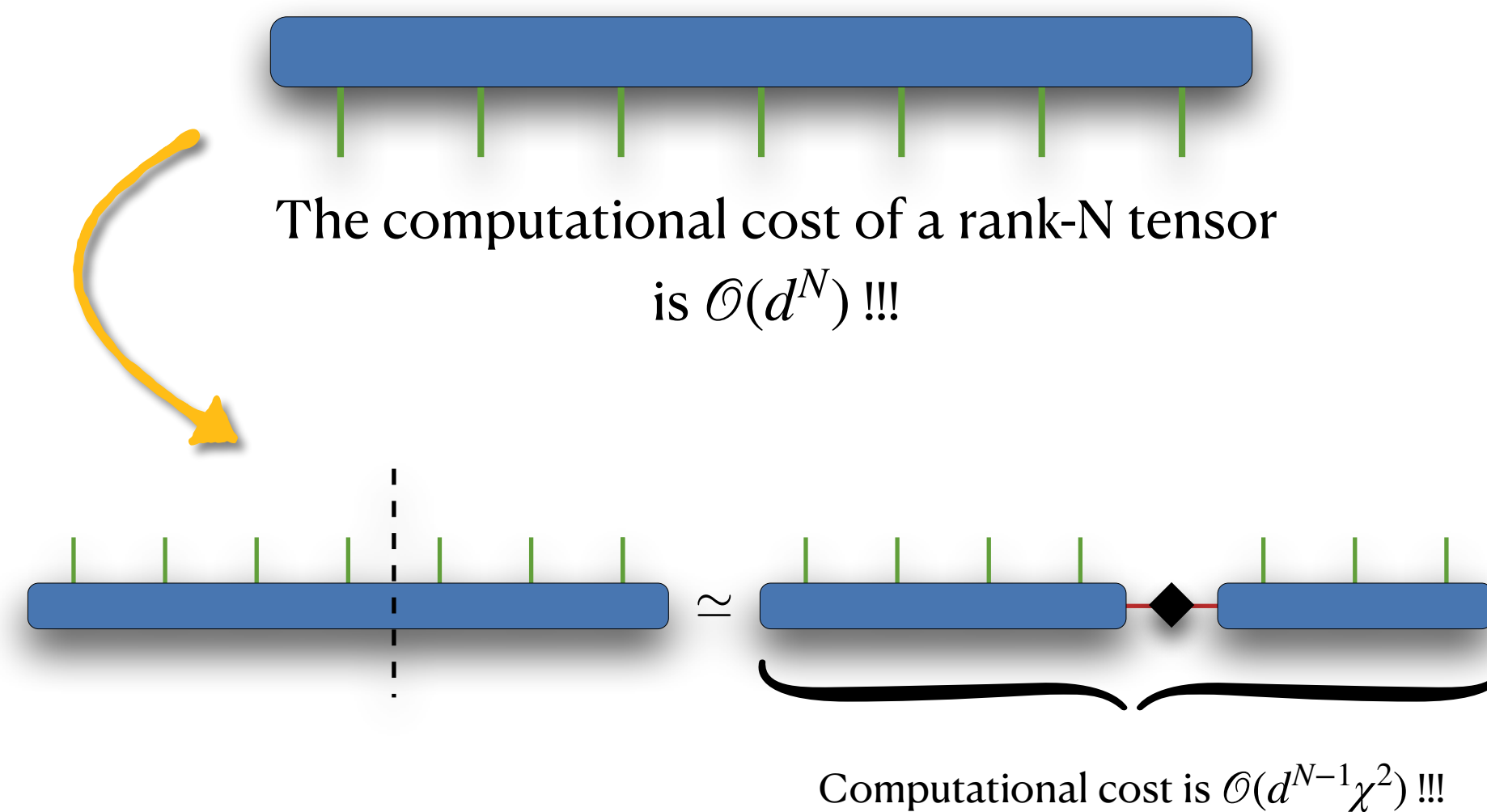
Tensor Networks: Origins

Tensor Diagram Notation



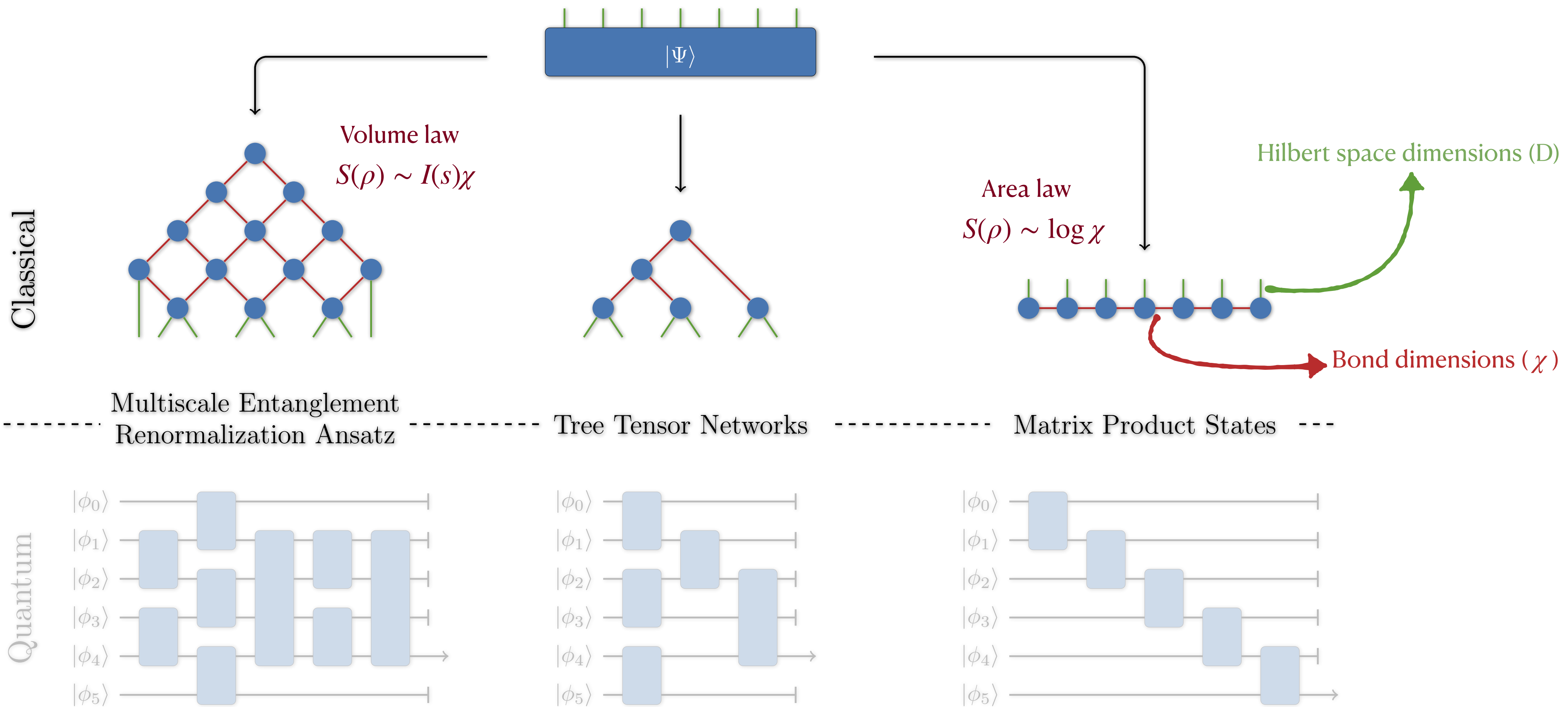
$$|\Psi\rangle = \sum_{\phi_1, \dots, \phi_n=0} \mathcal{W}_{\phi_1 \dots \phi_n} |\phi_1\rangle \otimes |\phi_2\rangle \otimes \dots \otimes |\phi_n\rangle$$

$$\forall |\phi_i\rangle \in \mathcal{H}^{\otimes 2^N} \rightarrow |\phi_i\rangle \in \{|\uparrow\rangle, |\downarrow\rangle\}$$



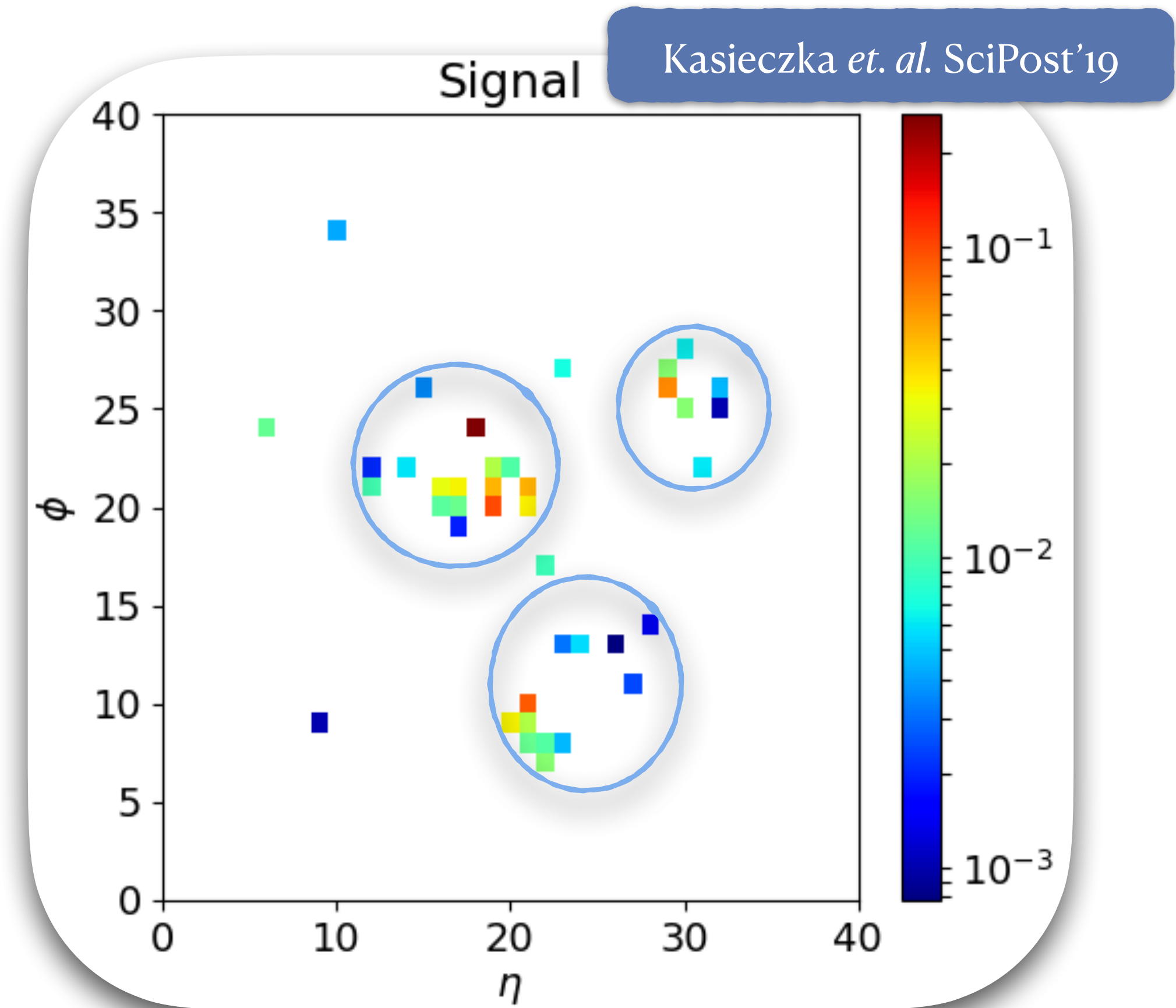
Types of Tensor Networks (some of them)

$$S(\rho) = \text{Tr}[\rho \log \rho]$$



Why TNs “might” perform well in such tasks?

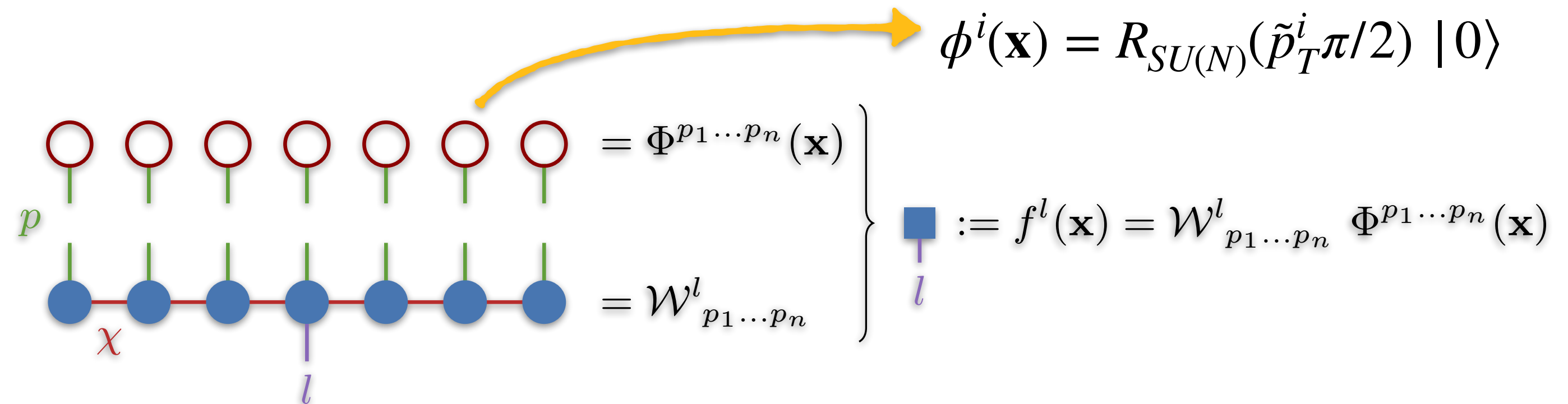
- The **correlation range** of a node in an MPS is bounded by its **bond dimension** and **distance between correlated sites**.
- Tensor Networks can capture **local** “anomalies”.
- We are dealing with sparse, locally correlated calorimeter pixels.



Matrix Product States for Classification

Sub-Outline

- ❖ How to embed the data?
- ❖ How to form a network?
- ❖ How to train the network?



$$\mathcal{L} = \frac{1}{N} \sum_{x \in \mathbf{x}^N} q^{\text{truth}} \log(p(x^{(i)}; \theta))$$

Or anything else you like to minimise...

$p(x^{(i)}; \theta) = |f^l(x^{(i)})|^2$

**No activation function!!
Everything is linear!!**

TNs intrinsically have "topological nonlinearity"

Traditionally NNs are trained with SGD, but MPS is trained with **Density Matrix Renormalisation Group Algorithm**

Stoudenmire, Schwab 1605.05775

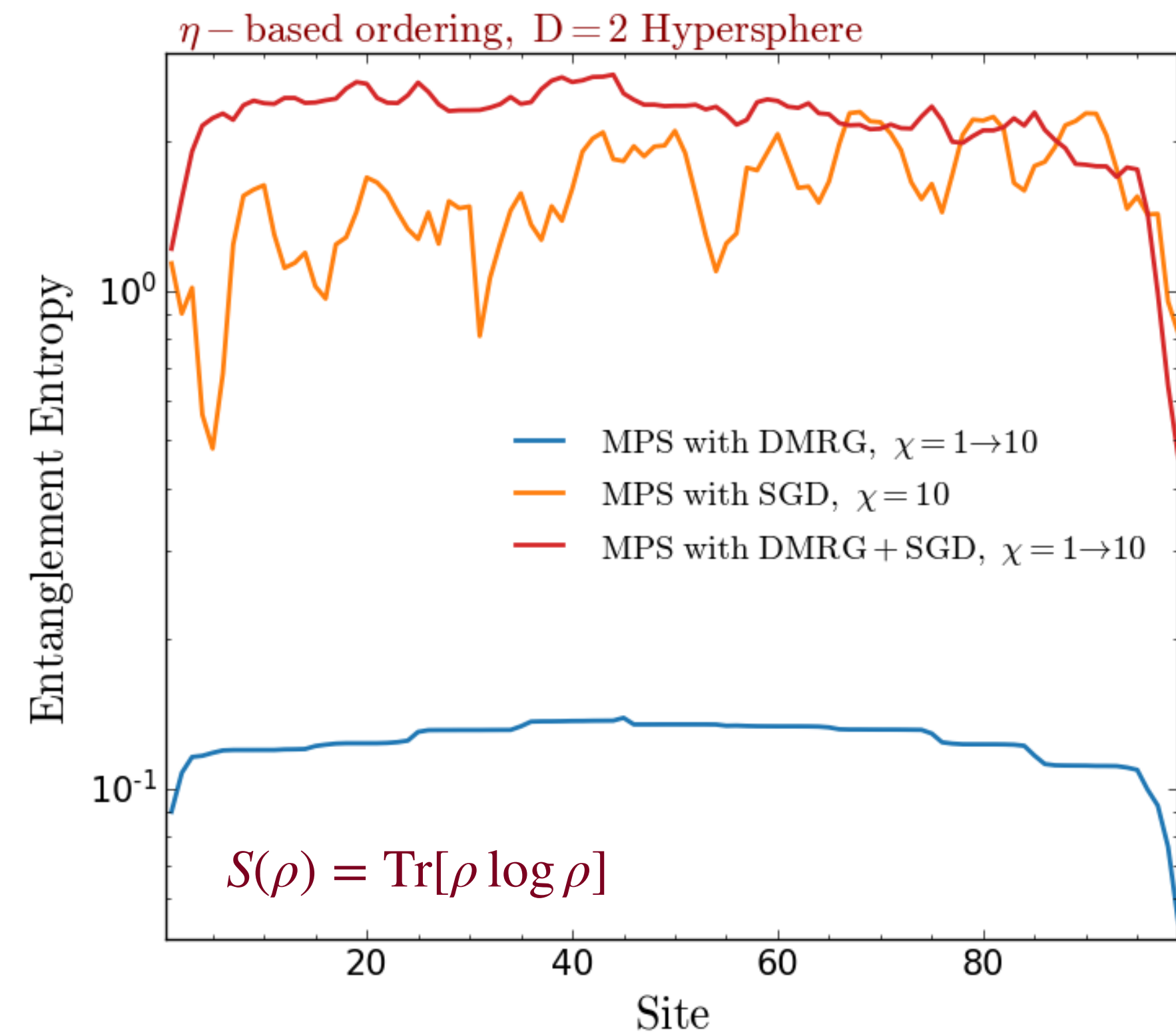
Novikov, Trofimov, Oseledets; 1605.03795

JYA, Spannowsky; JHEP '21, arXiv: 2106.08334

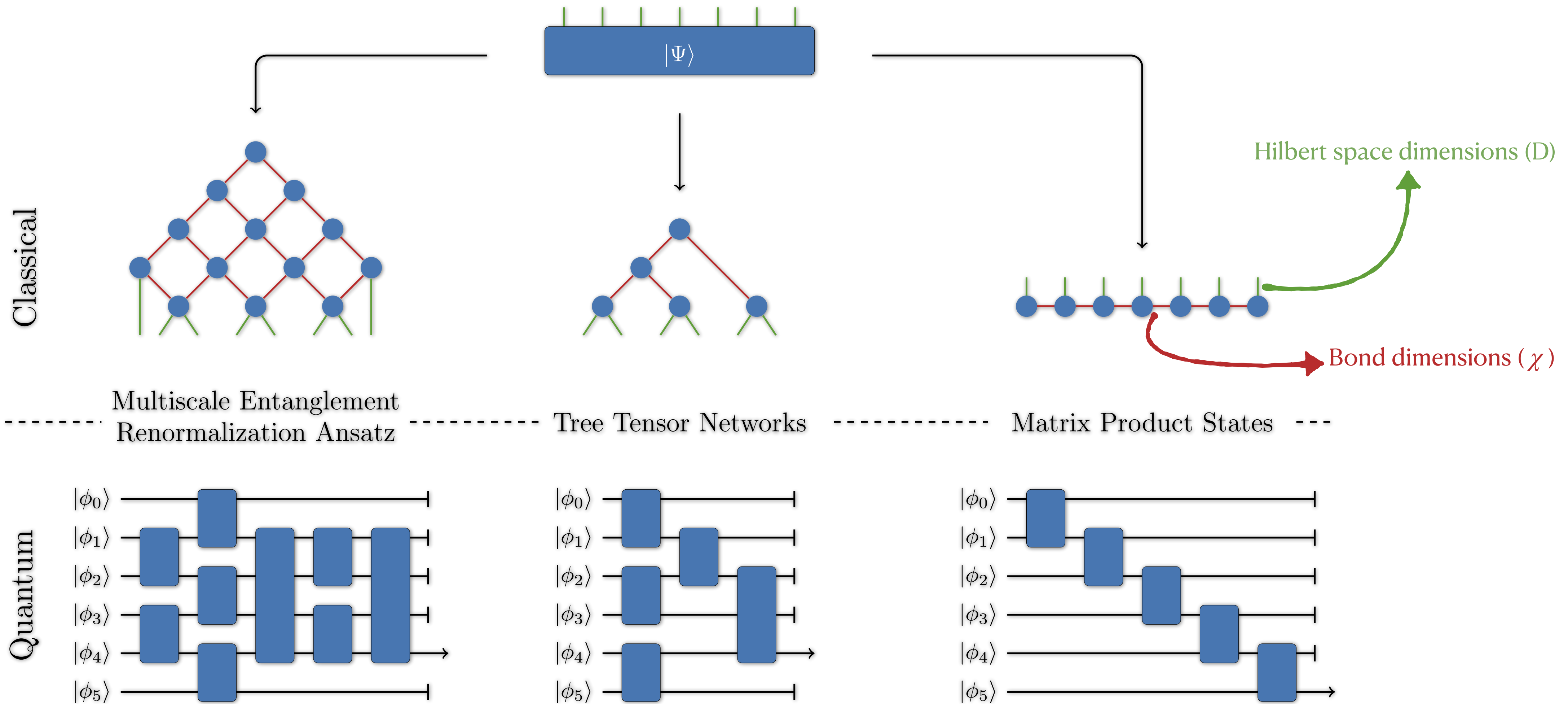
Why finding a quantitative measure is important?

JYA, Spannowsky; JHEP '21, arXiv: 2106.08334

- A **50% reduction in the number of pixels** used and a **91% reduction in the number of parameters** lead to the same classification quality!
- Understanding the network gives the ability to build better training algorithms.
- Scientific data is **largely sparse**; if we know where the information comes from, we can get rid of large amounts of data.
- **Suppress the noise** (and for pile-up mitigation to be confirmed)!

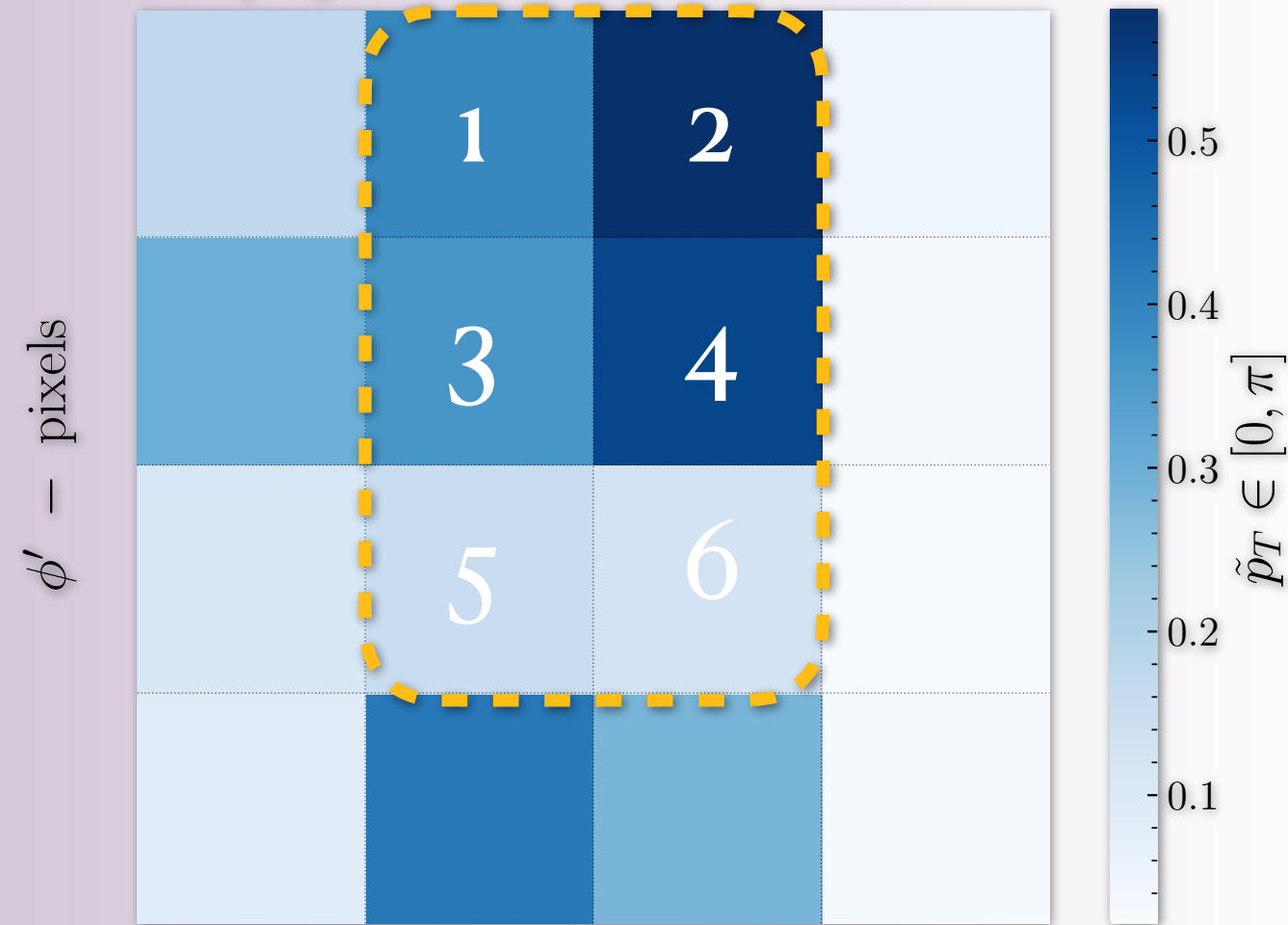


Types of Tensor Networks (some of them)

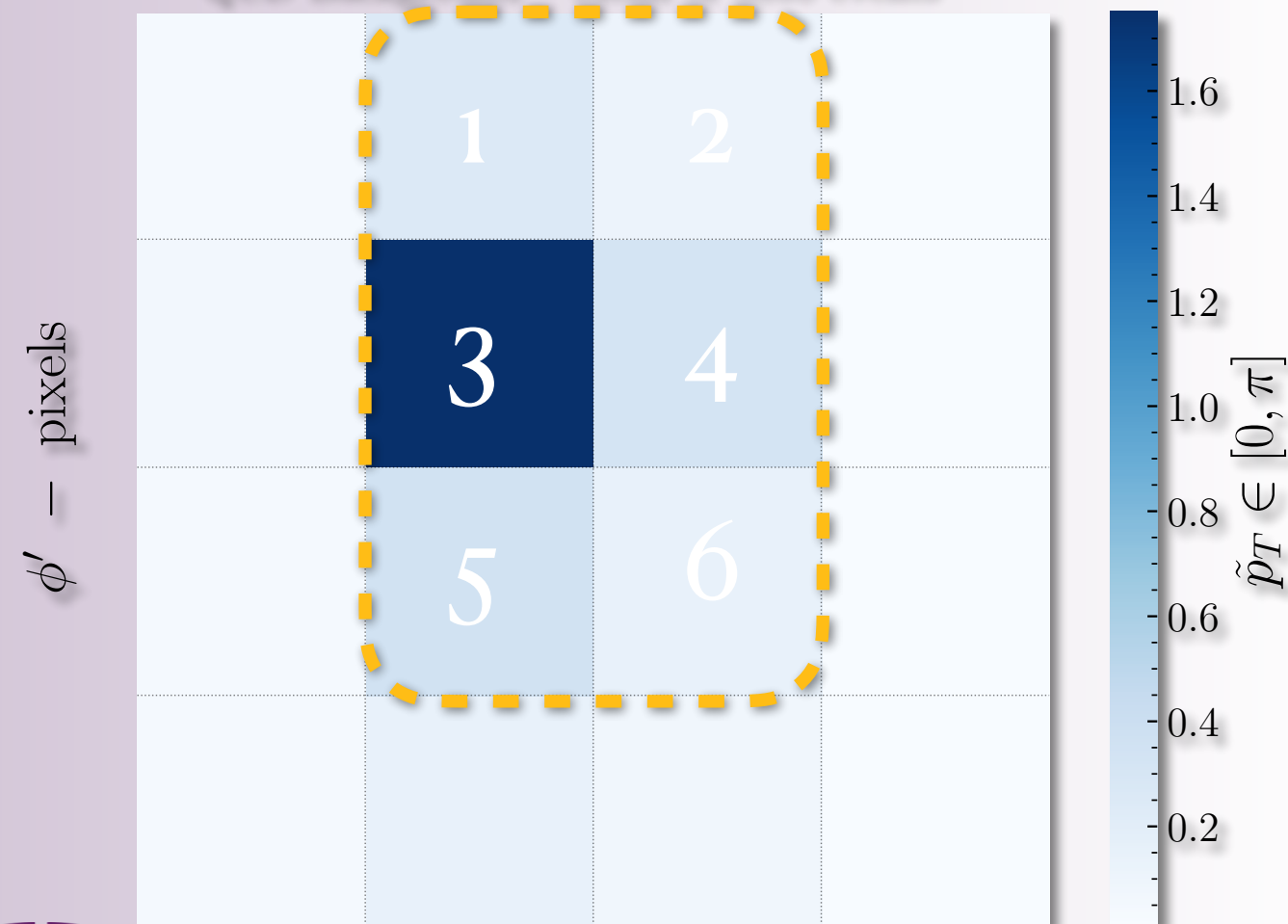


What can we gain if we adopt VQC?

Top Signal – Mean of 5000 events

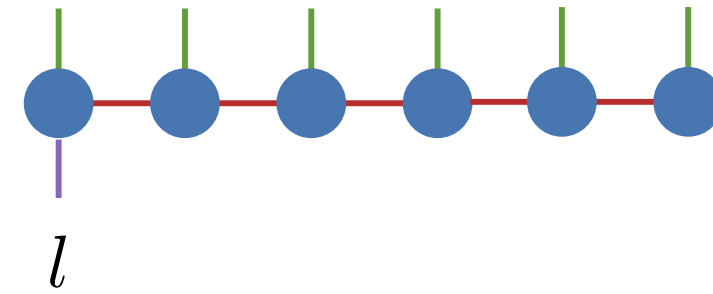


QCD Background – Mean of 5000 events

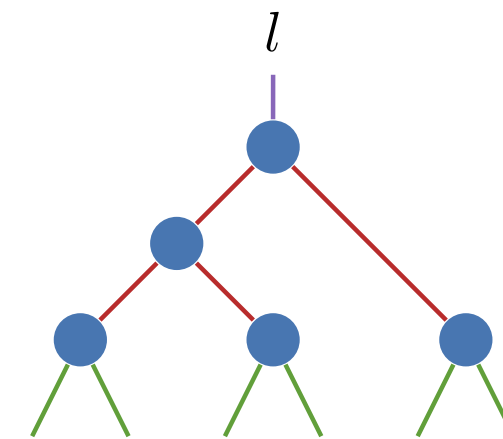


JYA, Spannowsky; PRA '22, arXiv: 2202.10471

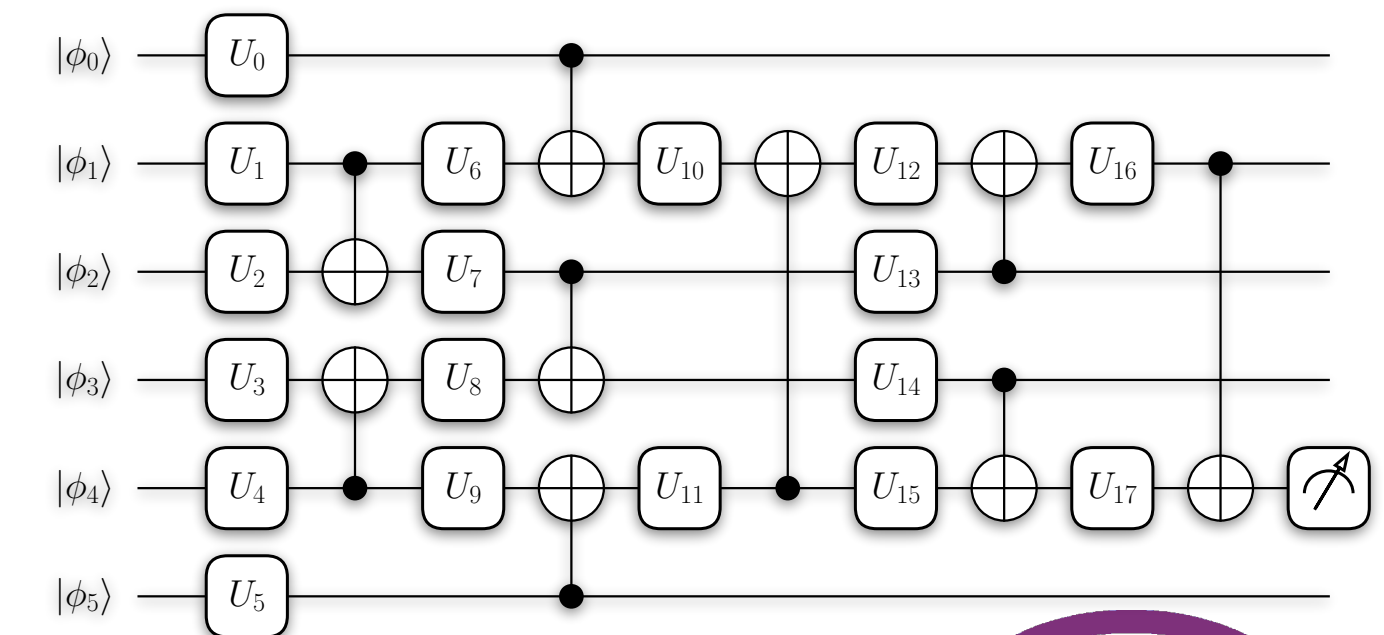
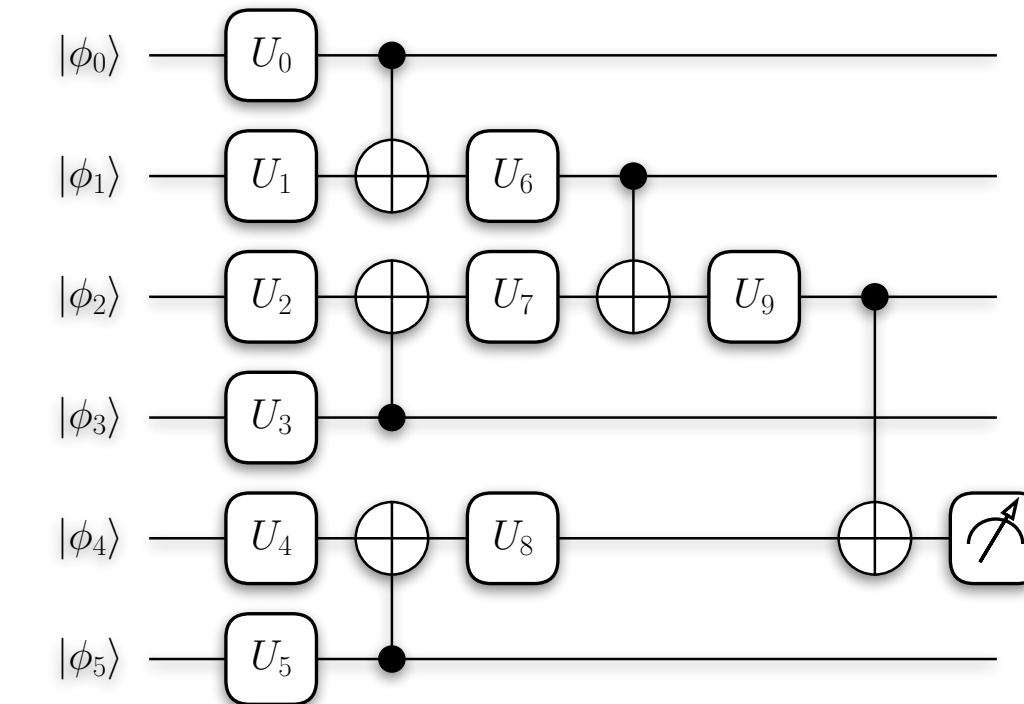
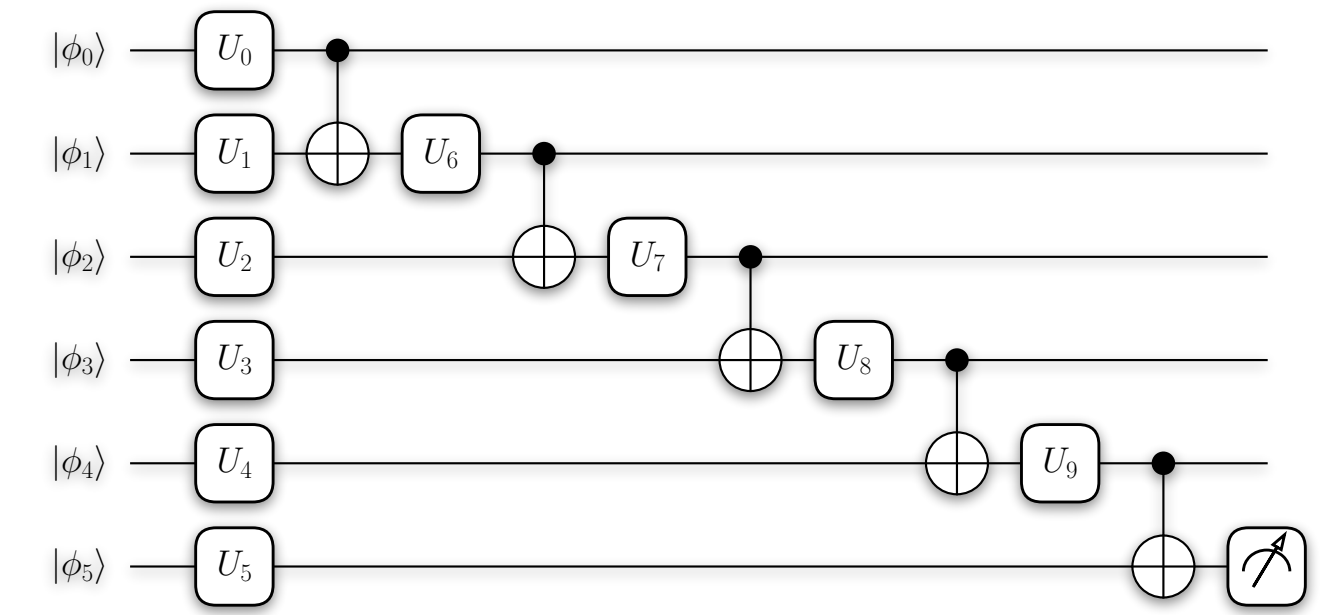
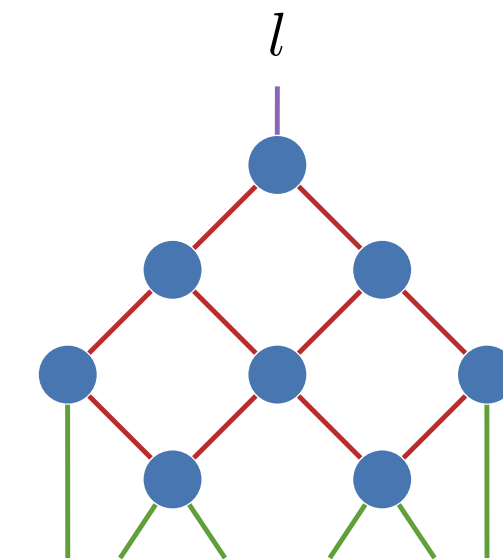
Matrix Product States



Tree Tensor Networks



Multiscale Entanglement Renormalisation Ansatz

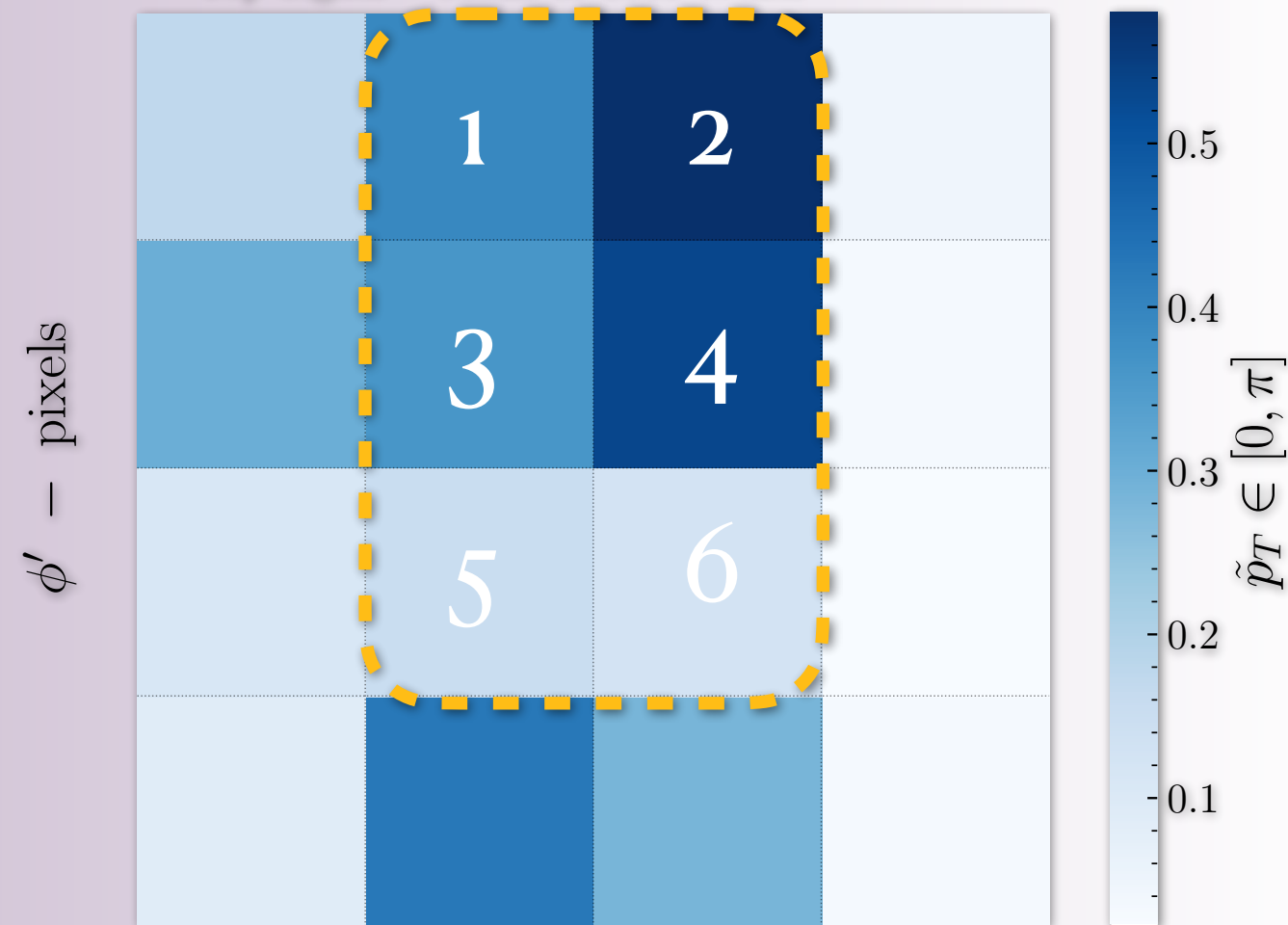


Jack Y. Araz



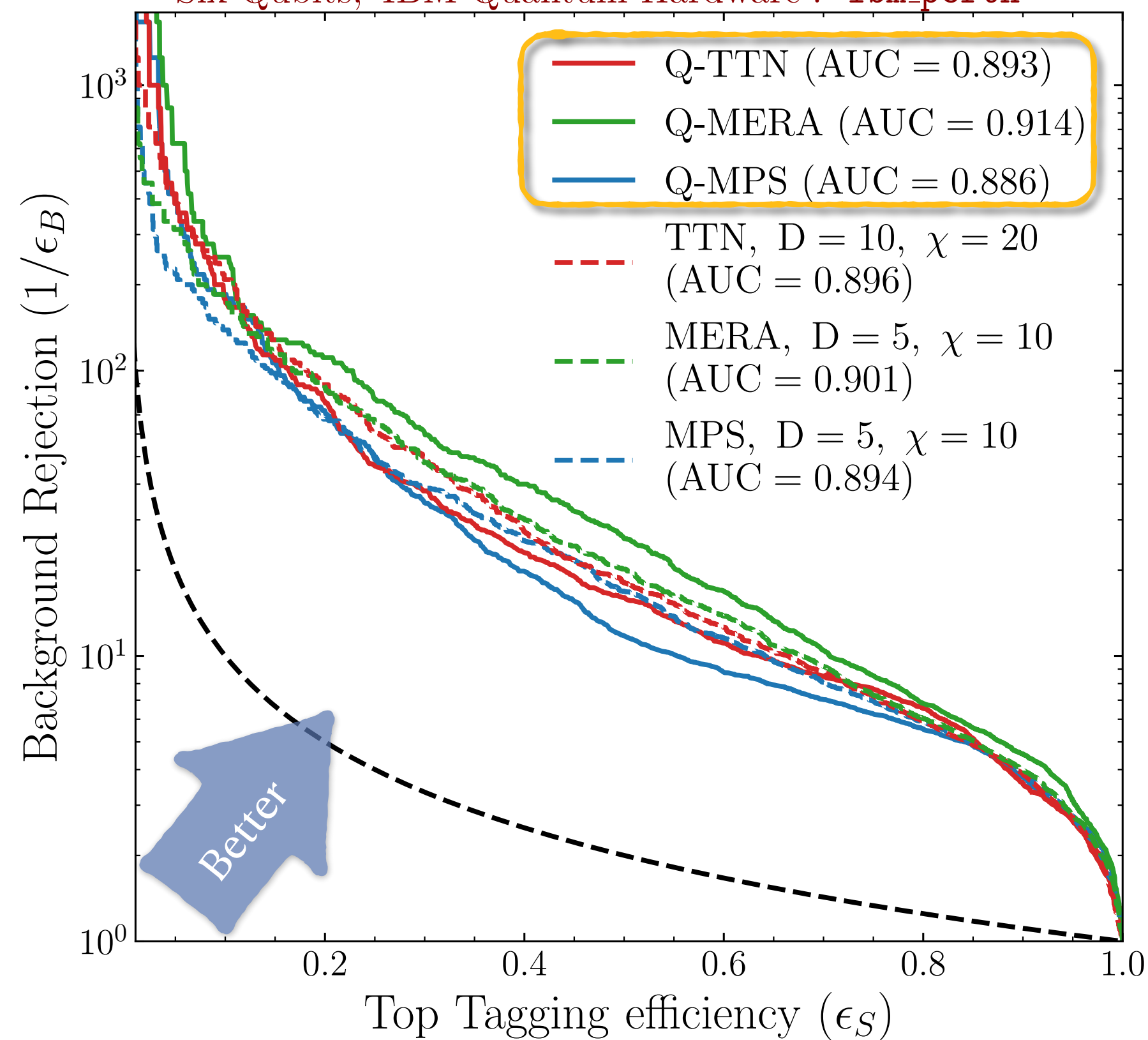
Experimenting with 6-Qubits

Top Signal – Mean of 5000 events

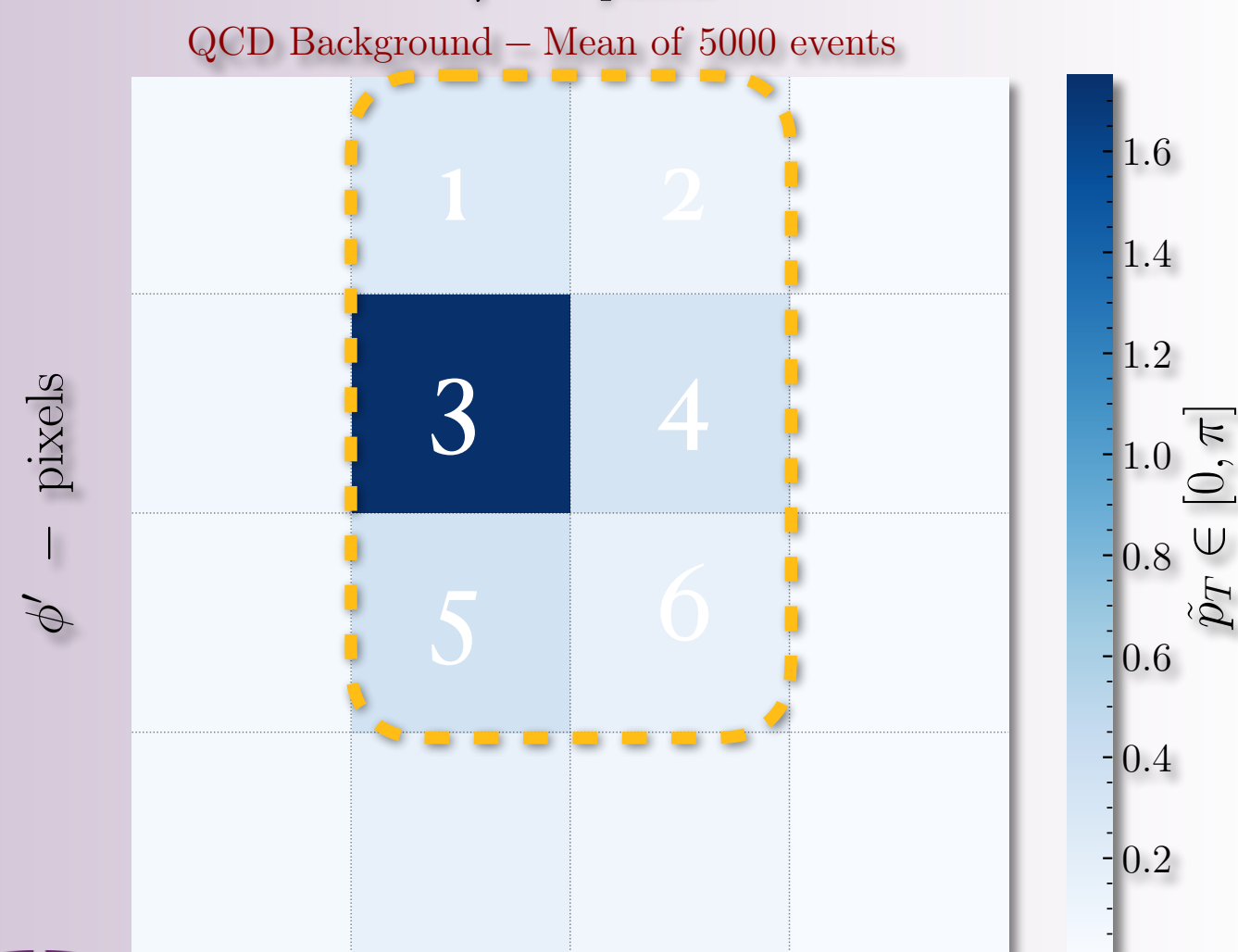


JYA, Spannowsky; PRA '22, arXiv: 2202.10471

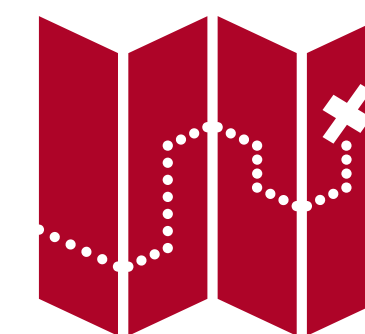
Six Qubits, IBM Quantum Hardware : *ibm_perth*



QCD Background – Mean of 5000 events



Ansatz	D	χ	# Parameters	AUC
TTN	2	5	235	0.755
	2	10	1320	0.803
	2	20	9040	0.849
	5	10	1950	0.873
	10	20	14800	0.896
MPS	2	5	230	0.811
	2	10	860	0.819
	2	20	3320	0.818
	5	10	2150	0.894
	5	10	18200	0.901
MERA	2	5	1225	0.850
	2	10	13400	0.840
	2	20	181600	0.848
	5	10	18200	0.901
Q-TTN	2		9	0.893
Q-MPS		9	0.886	
Q-MERA		17	0.914	

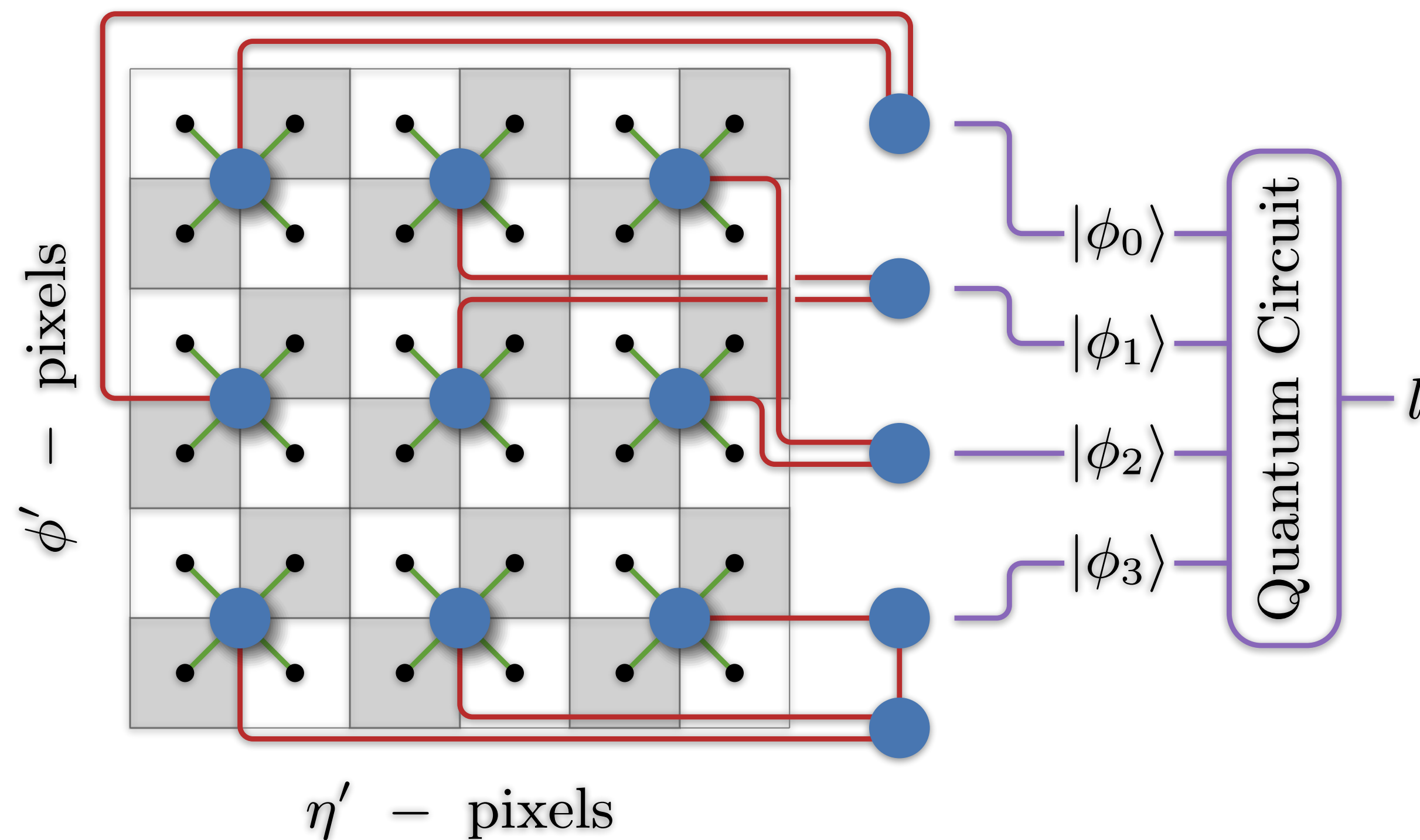


Analysing the Fisher information showed that the loss landscape for classical TNs becomes exponentially flat!



Ability to construct dynamic hybrid architectures!

JYA, Spannowsky; PRA '22, arXiv: 2202.10471



- ❖ Near-term quantum devices are quite limited; hence hybrid quantum-classical systems are essential
- ❖ Tensor Network nodes can be dynamically converted into qubits, as more will be available in the future!

Conclusion

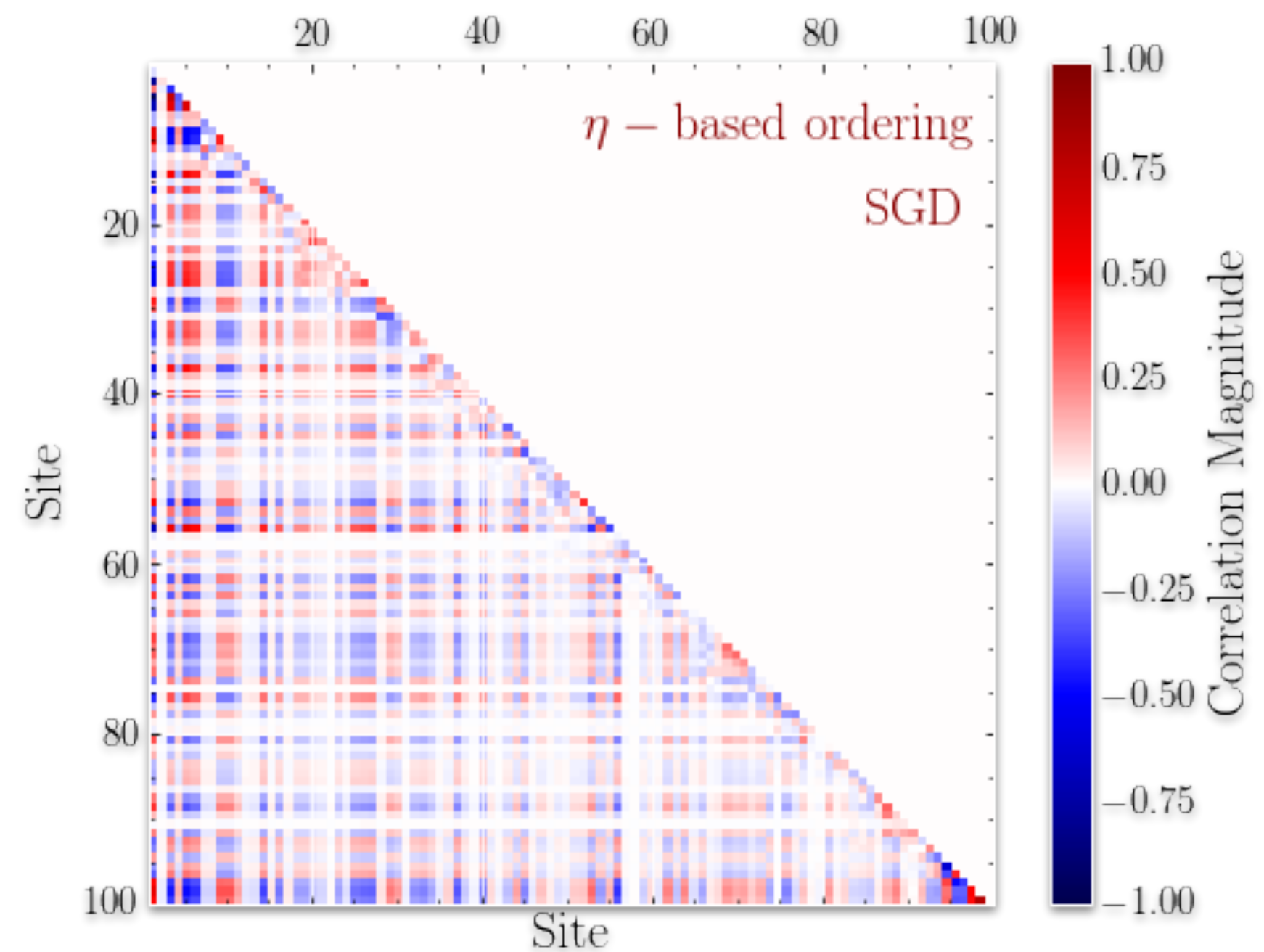
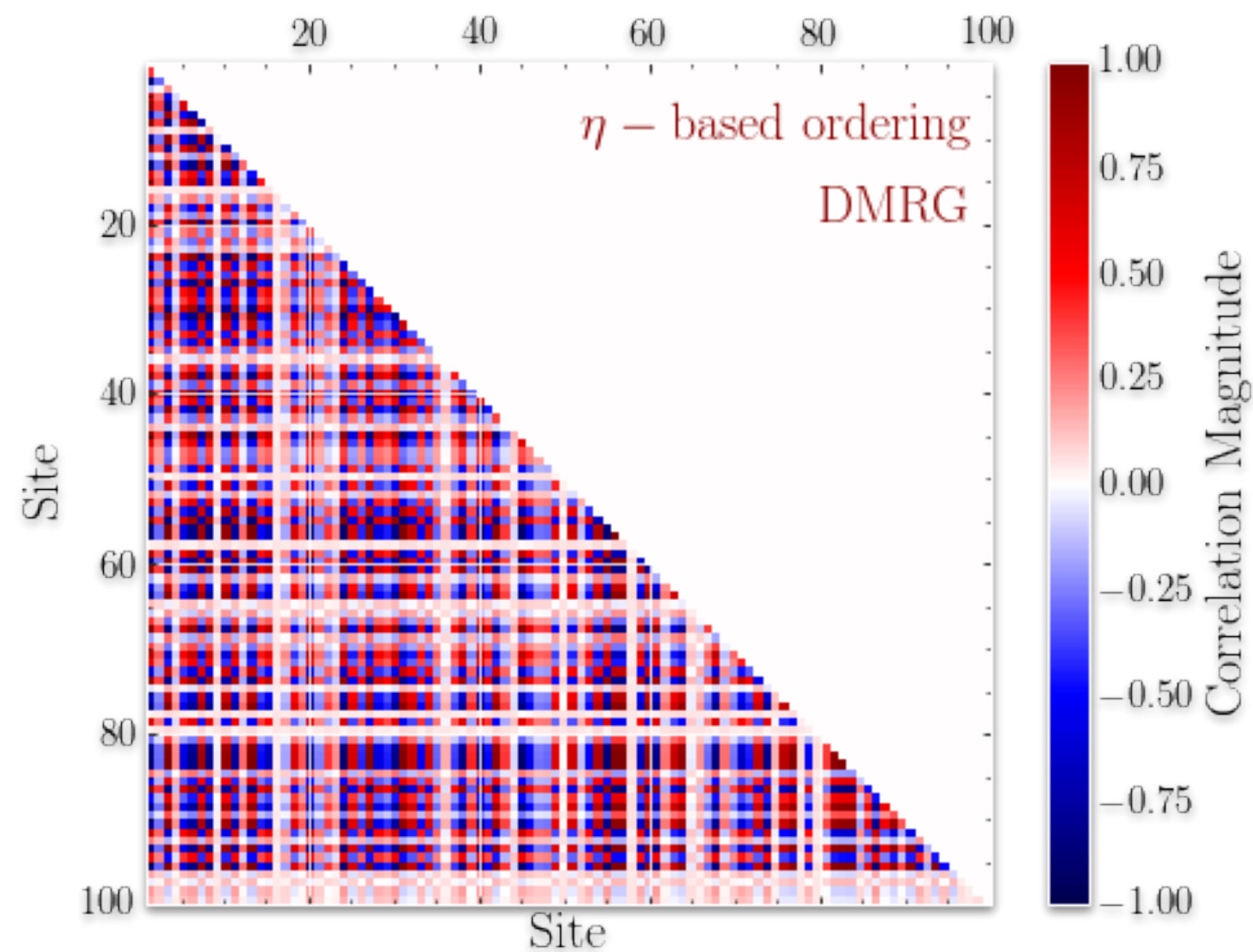
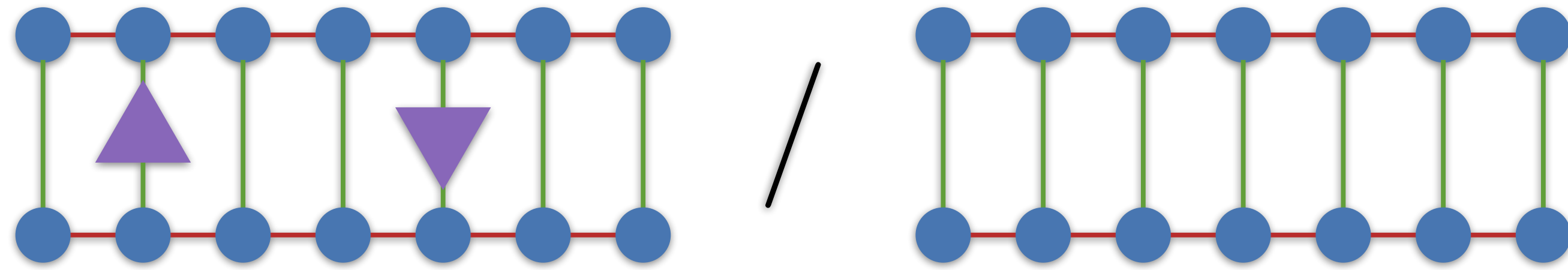
Conclusion

- ❖ The name of the game is **optimisation**. Techniques developed for field theory computations are easily transferable for ML applications!
- ❖ Designing **quantifiable measures** from the ansätze can allow us to improve training procedures and can be used for **feature selection**. (Thought: maybe helpful to lock on a symmetry during training?)
- ❖ Tensor Networks are the tool for the **near future** to understand quantum computing until the machinery is ready for more significant problems.

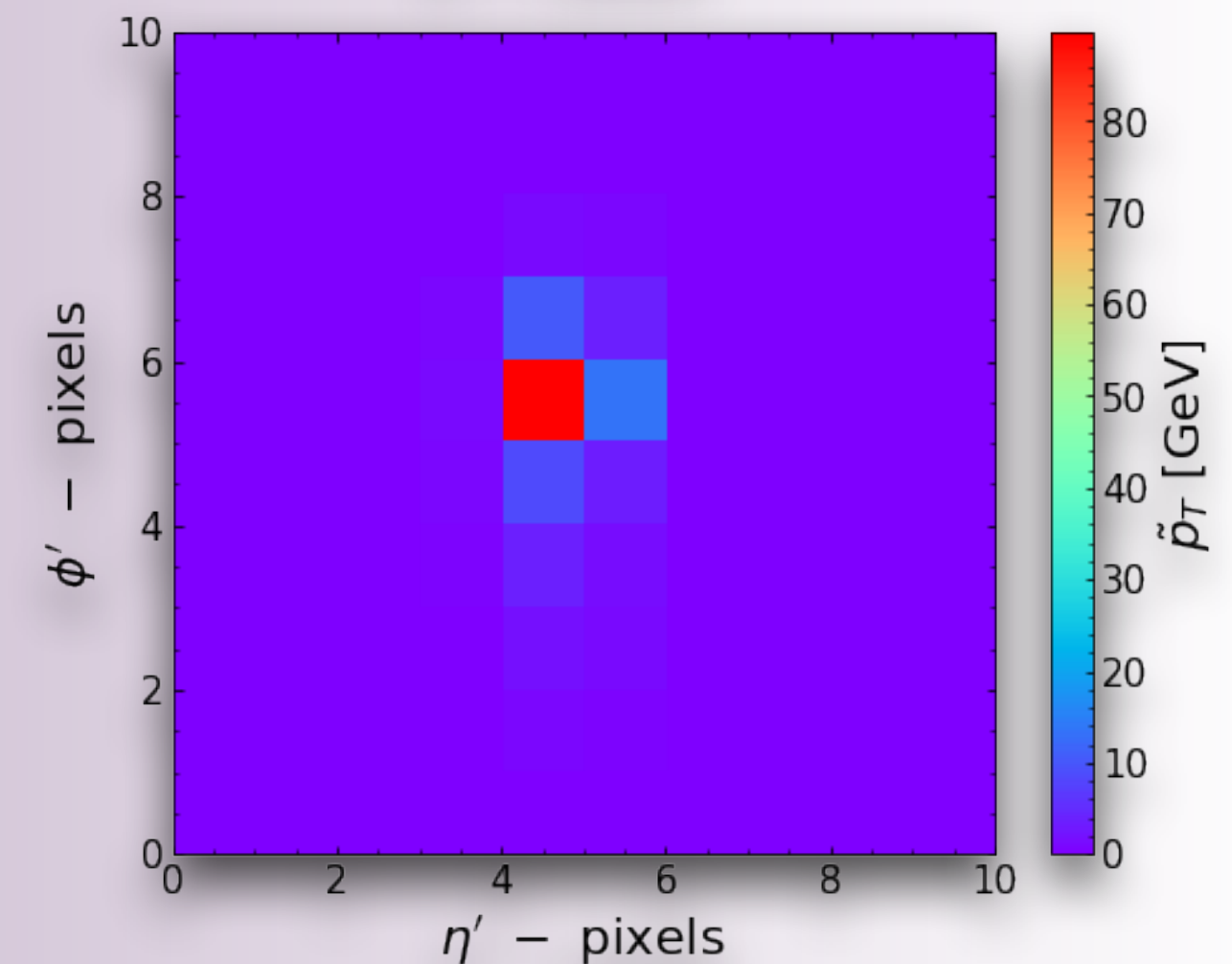
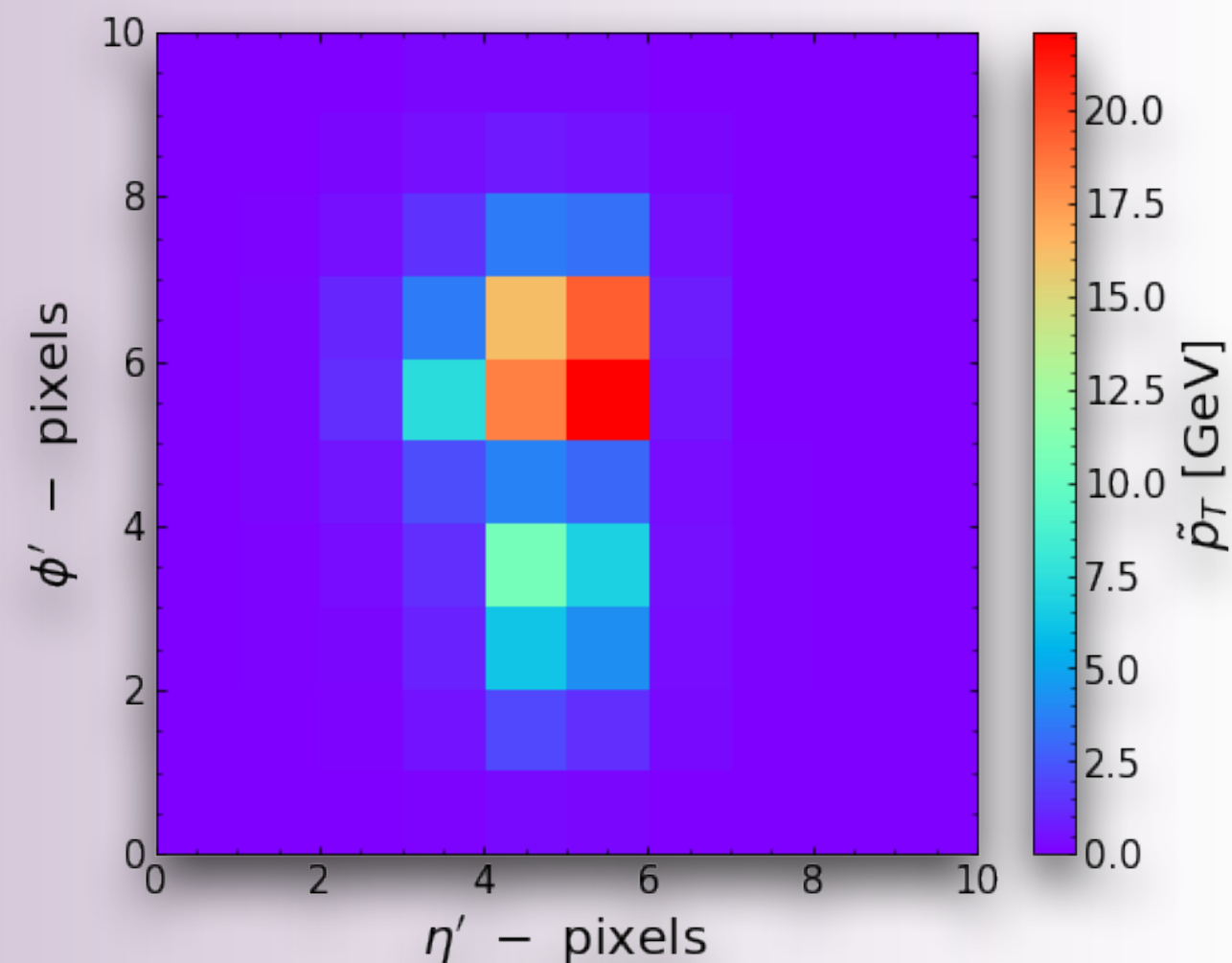
BACKUP

Correlations by SU(2) generators

$$C_{ij}^l = \frac{\langle \mathcal{W}^l | \mathcal{O}_i \mathcal{O}_j^\dagger | \mathcal{W}^l \rangle}{\langle \mathcal{W}^l | \mathcal{W}^l \rangle} =$$



Top Tagging through MPS

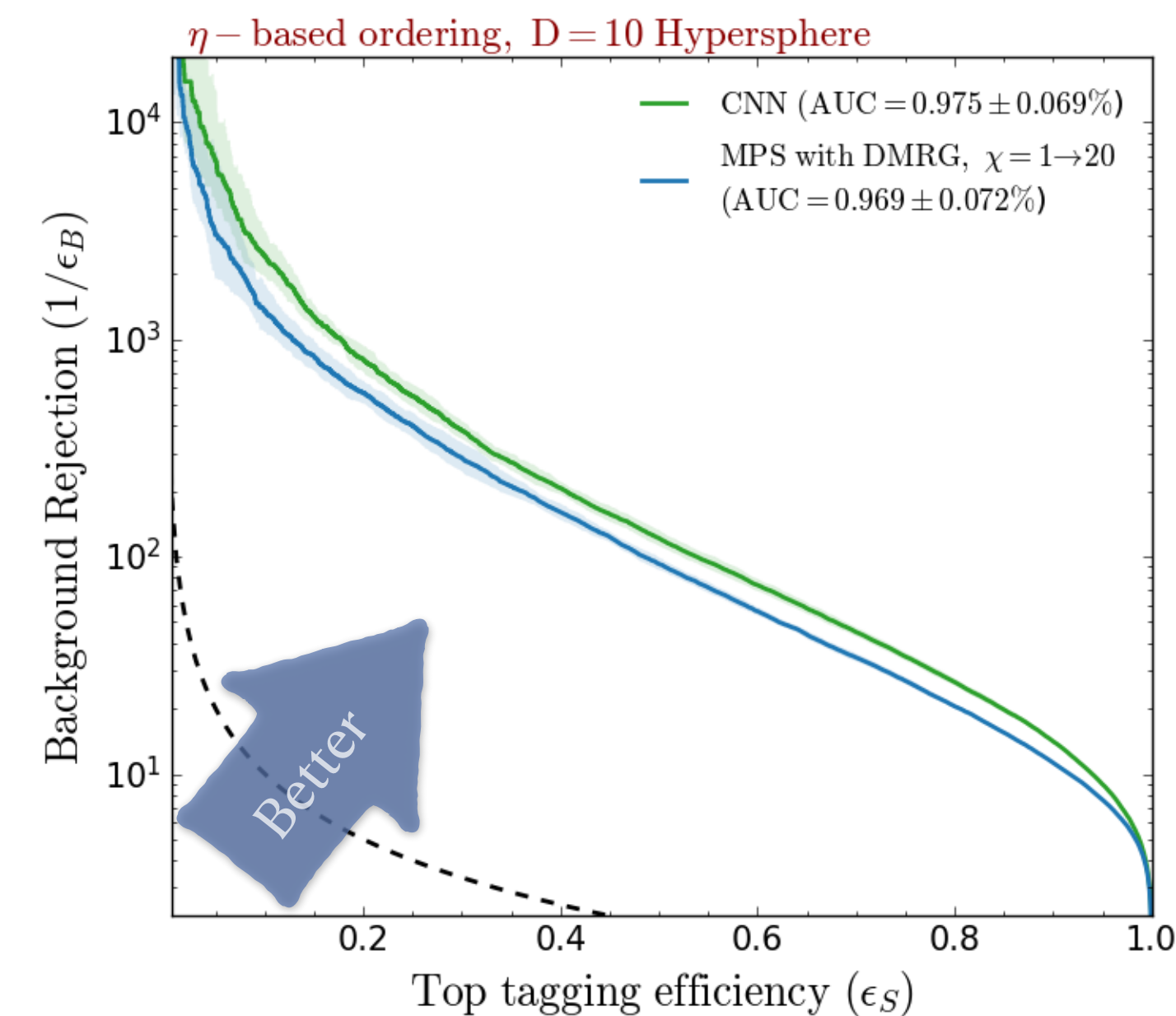
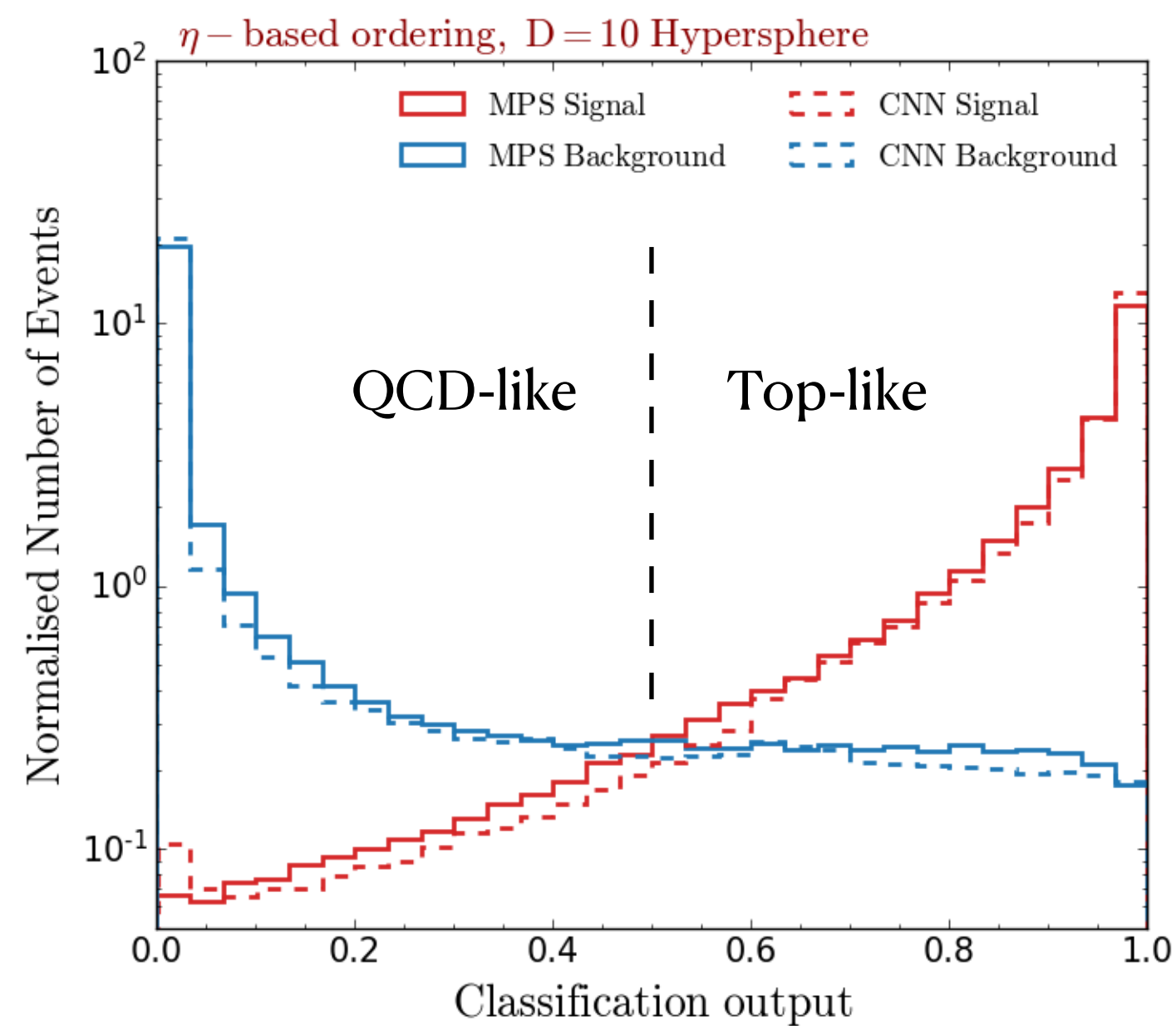


Assumptions & Requirements

- ❖ No prior entanglement/correlation between pixels
- ❖ Network is a Born Machine \rightarrow square of the wave-function gives the probability of the classification.
- ❖ Maximum bond dimension that network can get is 20.

CNN architecture from:

JYA, Spannowsky; JHEP '21

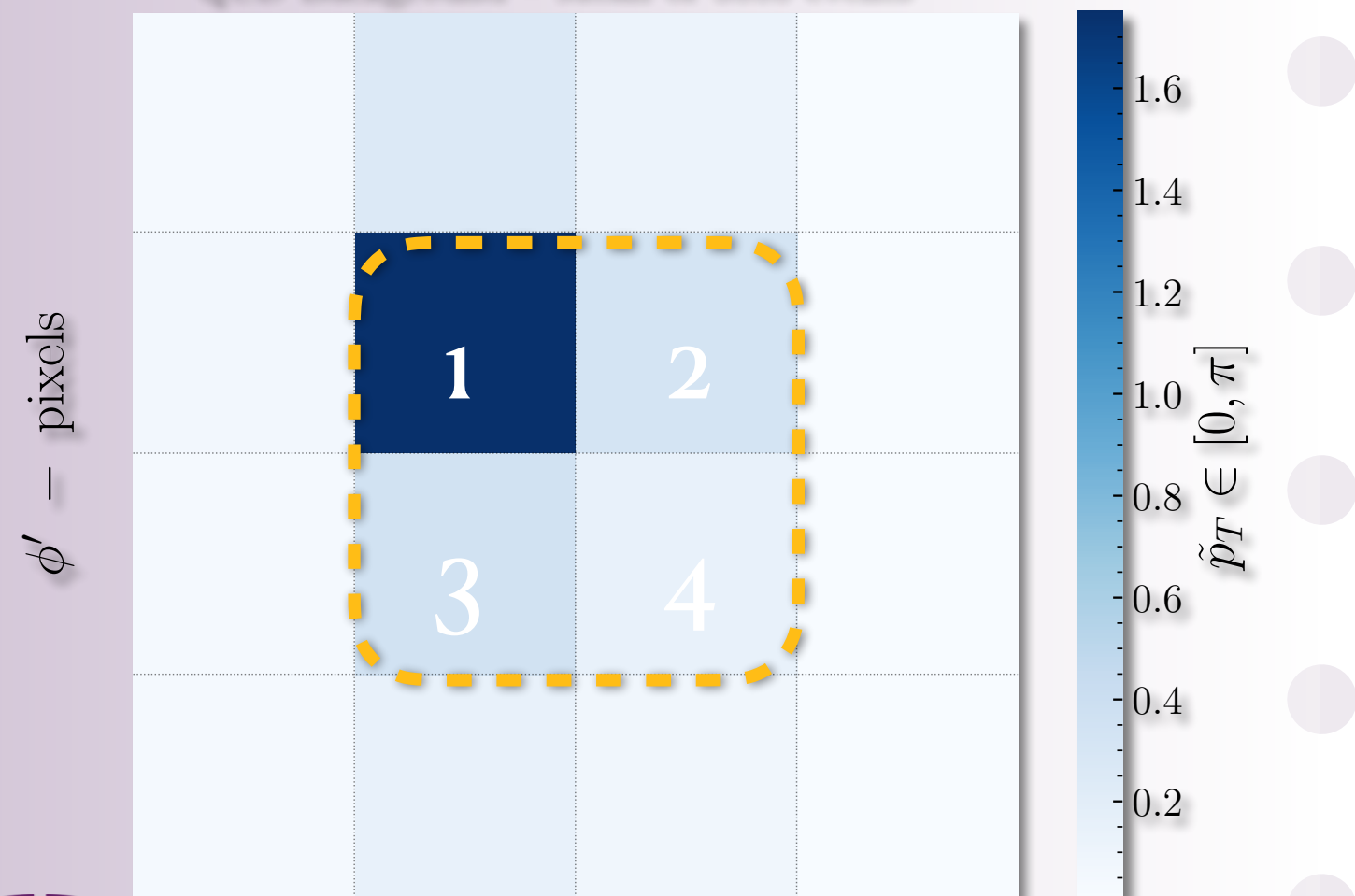


Experimenting with 4-Qubits

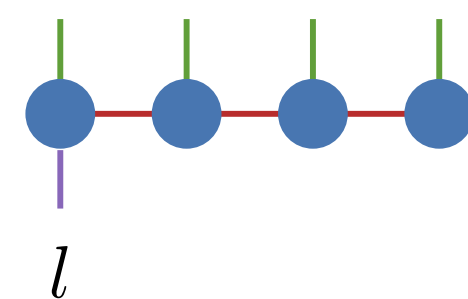
Top Signal – Mean of 5000 events



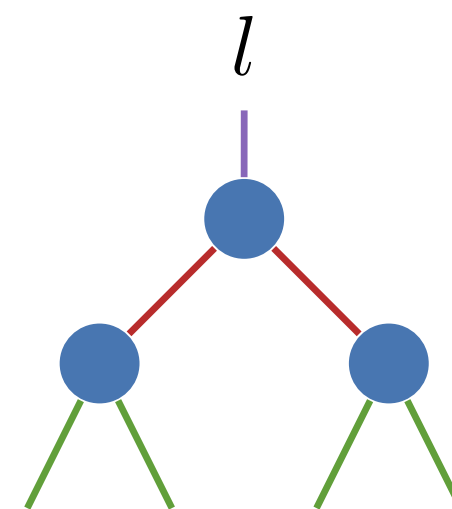
QCD Background – Mean of 5000 events



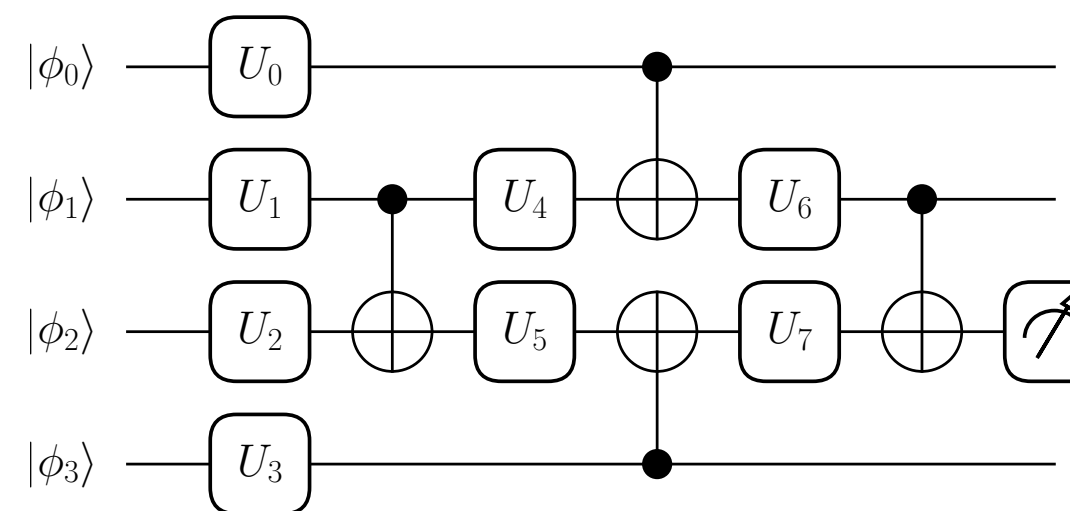
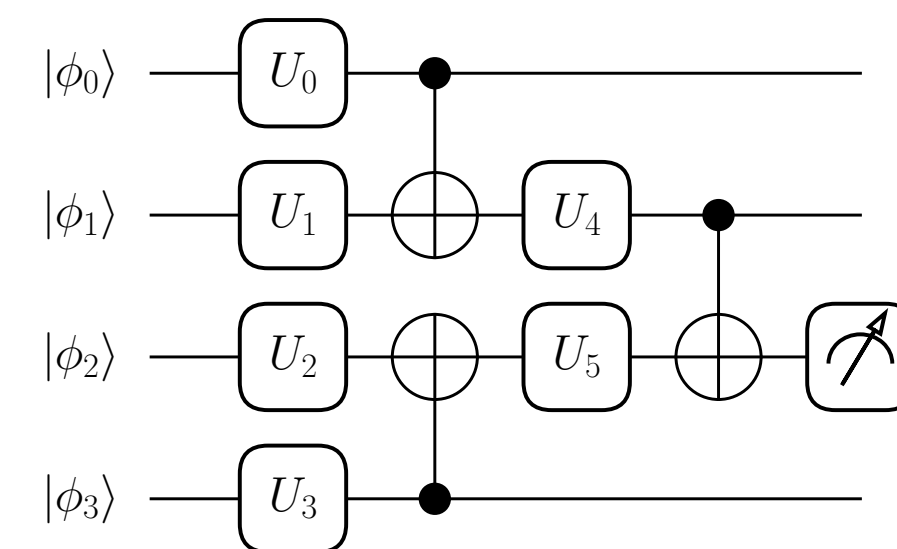
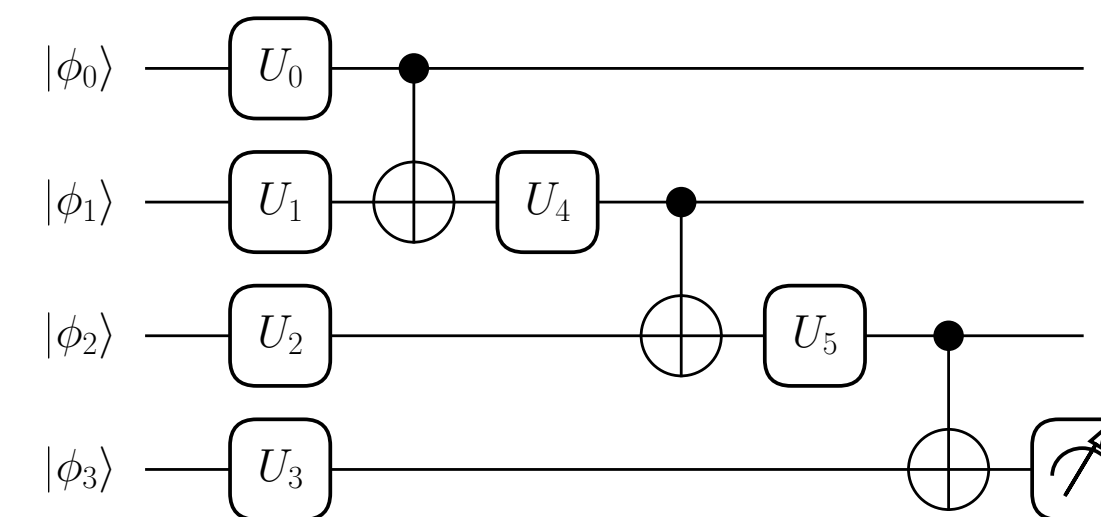
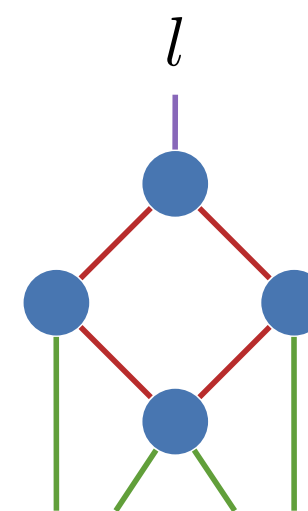
Matrix Product States



Tree Tensor Networks

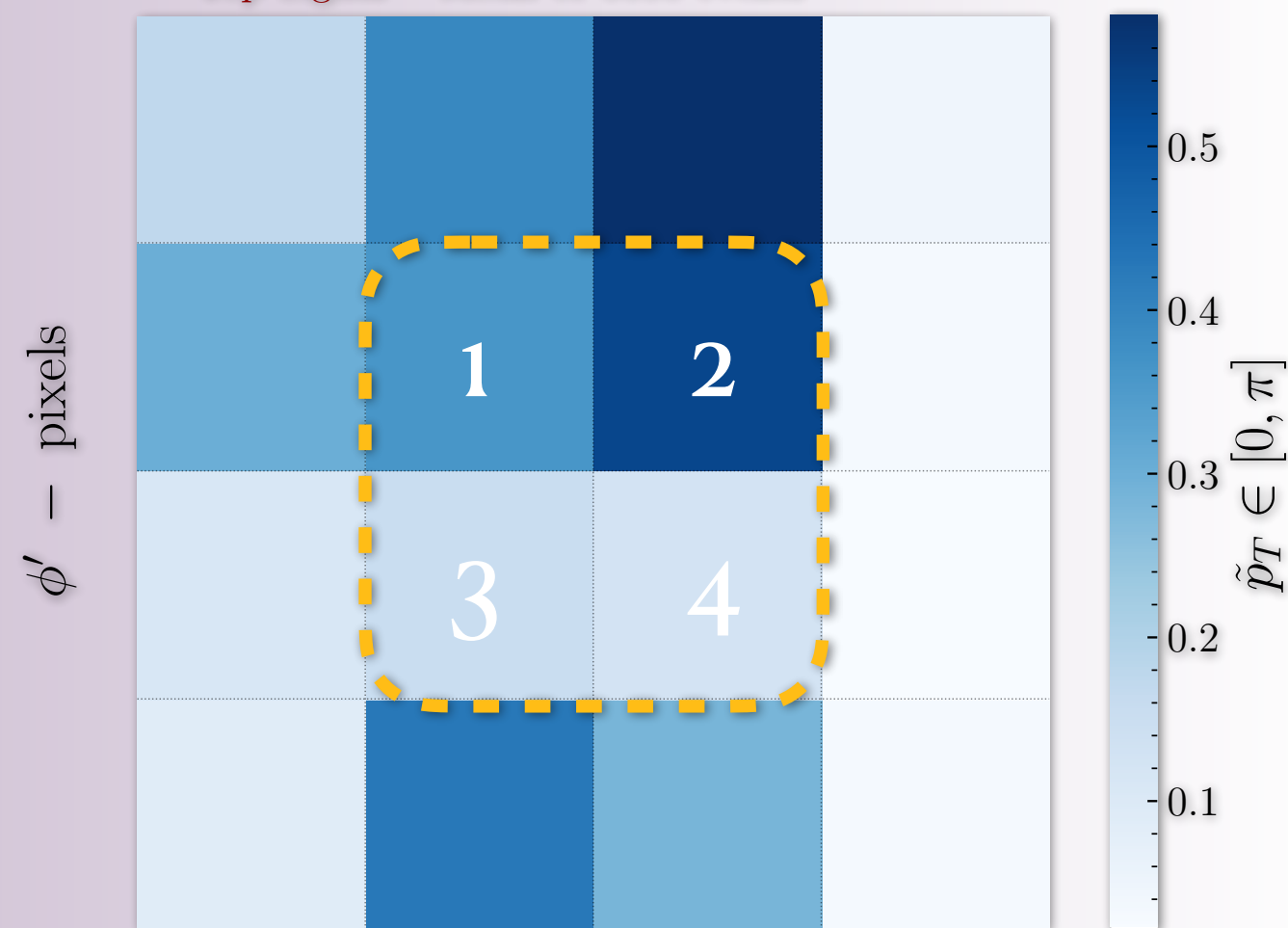


Multiscale Entanglement Renormalisation Ansatz



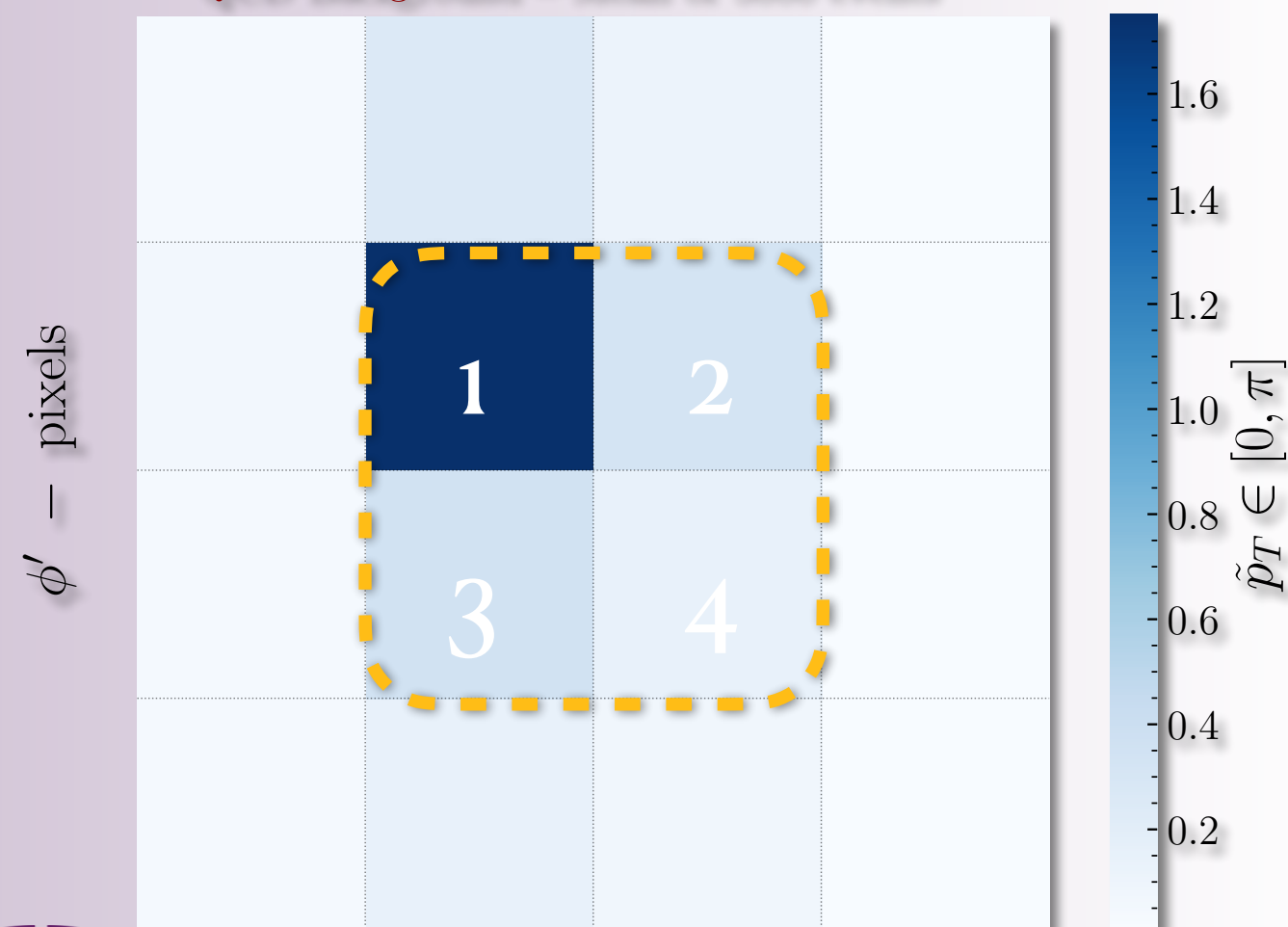
Experimenting with 4-Qubits

Top Signal – Mean of 5000 events

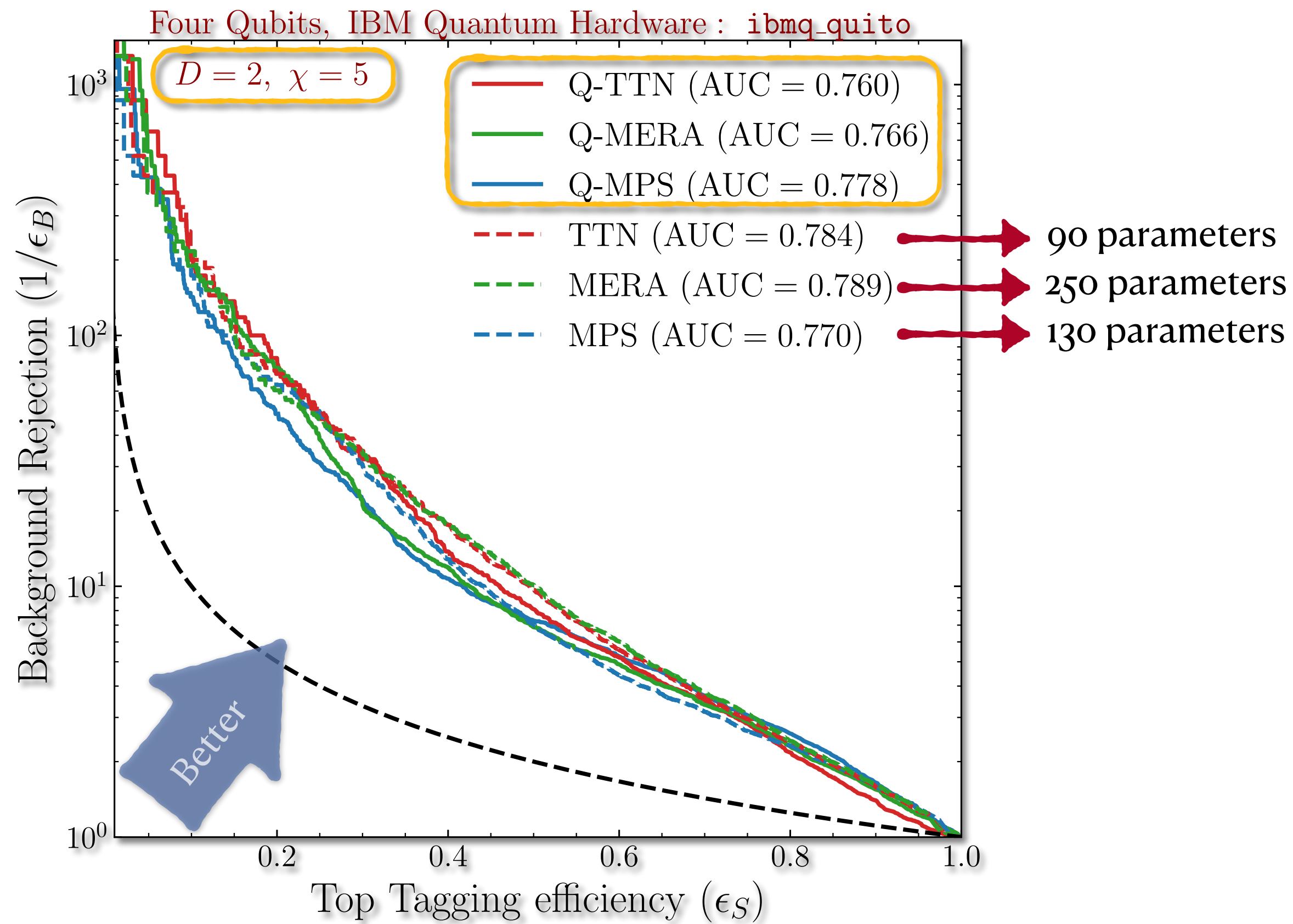


η' – pixels

QCD Background – Mean of 5000 events



η' – pixels



Singular Value Decomposition

$$\begin{matrix}
 m \times n & & m \times k & & k \times l & & l \times n \\
 \left[\begin{array}{cccc} \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \end{array} \right] & = & \left[\begin{array}{cccc} \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \end{array} \right] & \left[\begin{array}{ccc} \bullet & & \\ & \bullet & \\ & & \bullet \end{array} \right] & \left[\begin{array}{cccc} \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \end{array} \right] \\
 \mathbf{M} & = & \mathbf{U} & \mathbf{S} & \mathbf{V}^\dagger
 \end{matrix}$$

Orthogonal singular column vectors

Positive definite singular values in descending order

Orthogonal singular row vectors

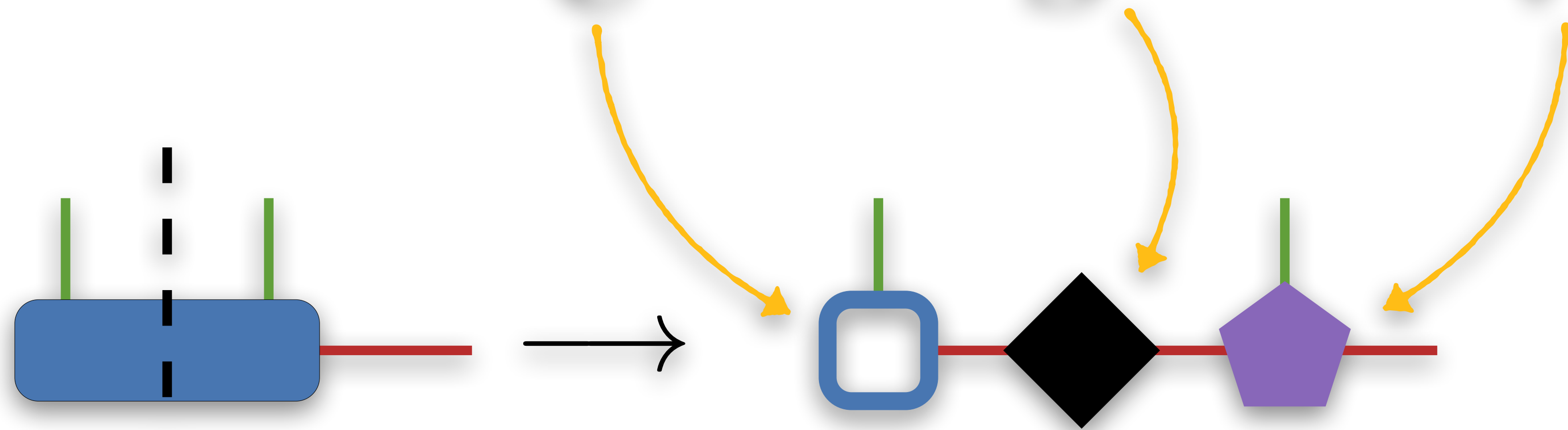
By changing the number of singular values one can change the accuracy of the decomposition!!!

$$\mathbf{S} = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$$

λ_i also known as Schmidt values

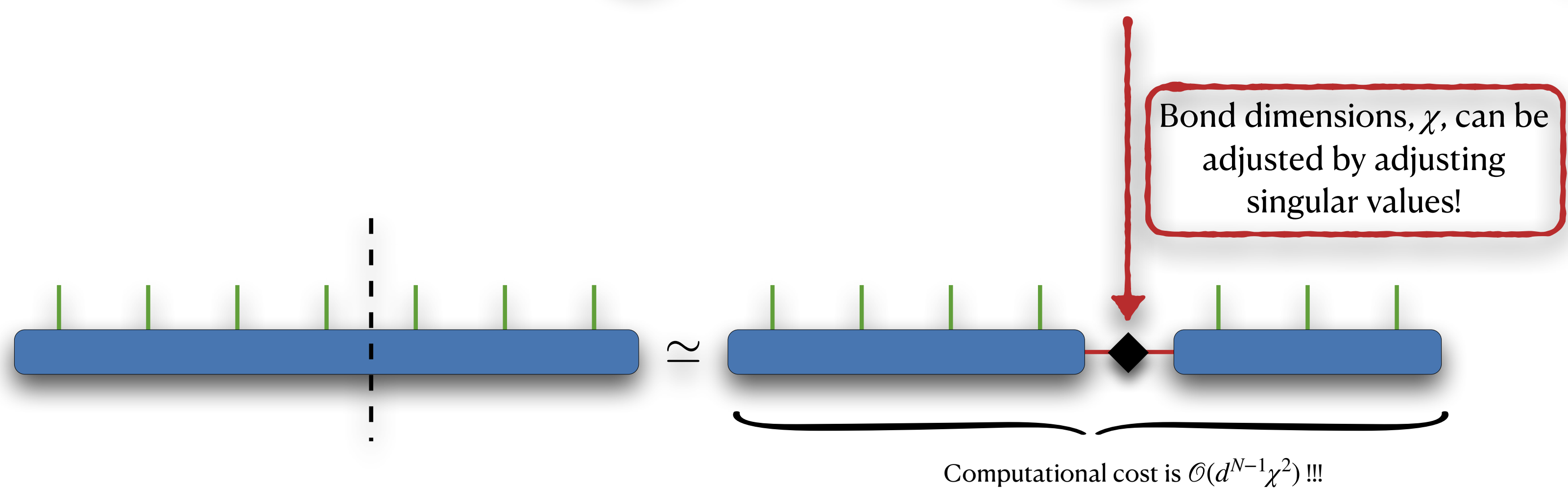
Singular Value Decomposition

$$\begin{matrix}
 m \times n & & m \times k & & k \times l & & l \times n \\
 \left[\begin{array}{cccc} \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \end{array} \right] & = & \left[\begin{array}{cccc} \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \end{array} \right] & \left[\begin{array}{cc} \bullet & \\ & \bullet \\ & \\ & \bullet \end{array} \right] & \left[\begin{array}{cccc} \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \end{array} \right] \\
 \mathbf{M} & = & \mathbf{U} & \mathbf{S} & \mathbf{V}^\dagger
 \end{matrix}$$



Singular Value Decomposition

$$\begin{matrix}
 m \times n & & m \times k & & k \times l & & l \times n \\
 \left[\begin{array}{cccc} \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \end{array} \right] & = & \left[\begin{array}{cccc} \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \end{array} \right] & \left[\begin{array}{cc} \bullet & \\ & \bullet \end{array} \right] & \left[\begin{array}{cccc} \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \end{array} \right] \\
 \mathbf{M} & = & \mathbf{U} & \mathbf{S} & \mathbf{V}^\dagger
 \end{matrix}$$



Matrix Product States for Classification

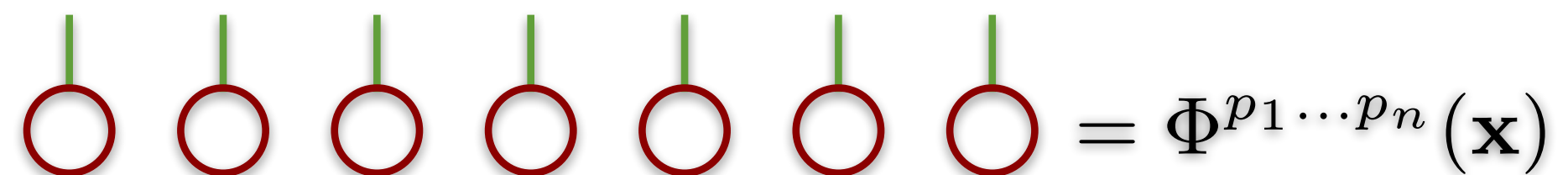
Sub-Outline

- How to embed the data?
- How to form a network?
- How to train the network?

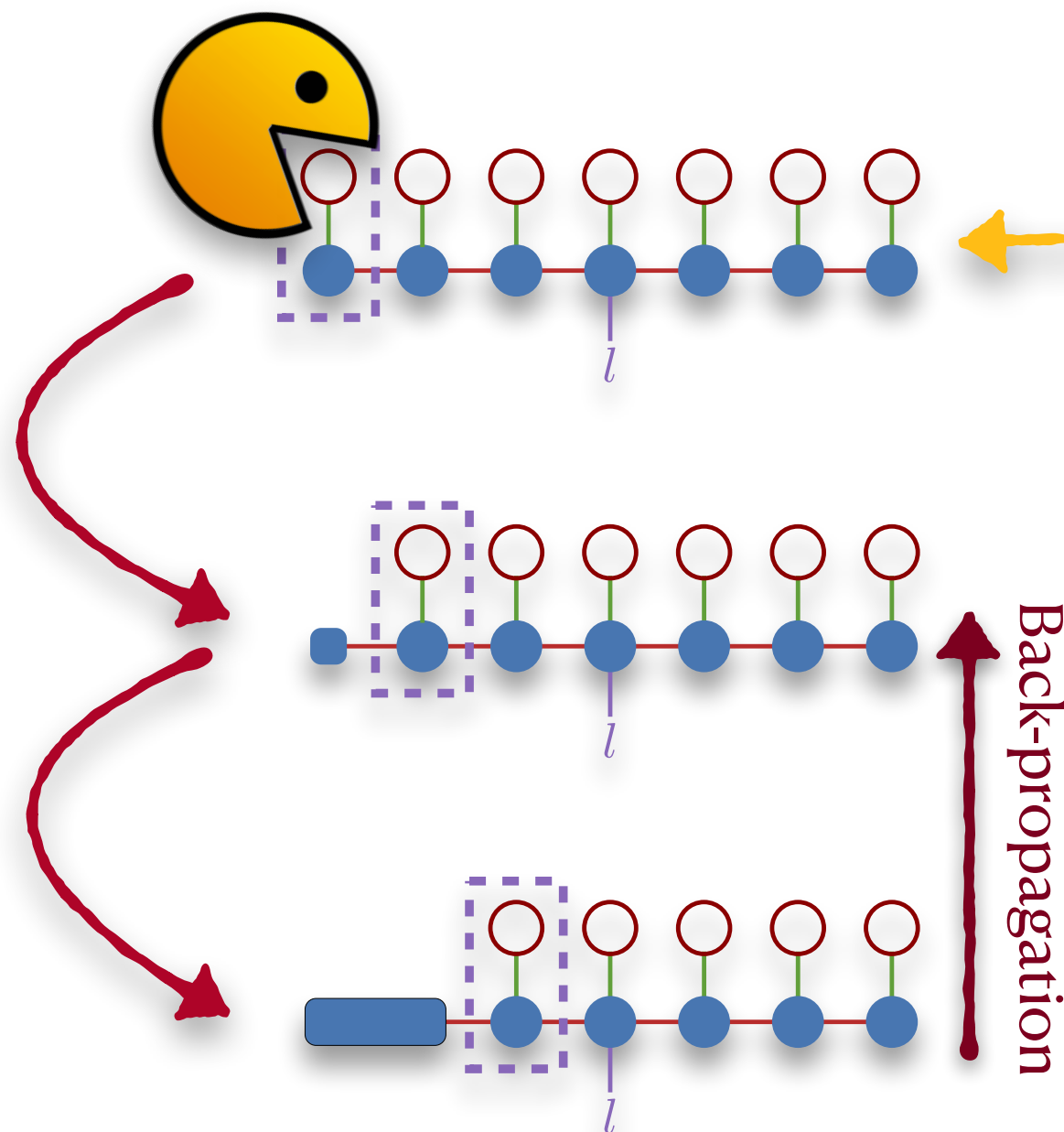
Data Embedding

$$\Phi^{p_1 \dots p_n}(\mathbf{x}) = \phi^{p_1}(x_1) \otimes \phi^{p_2}(x_2) \otimes \dots \otimes \phi^{p_n}(x_n)$$

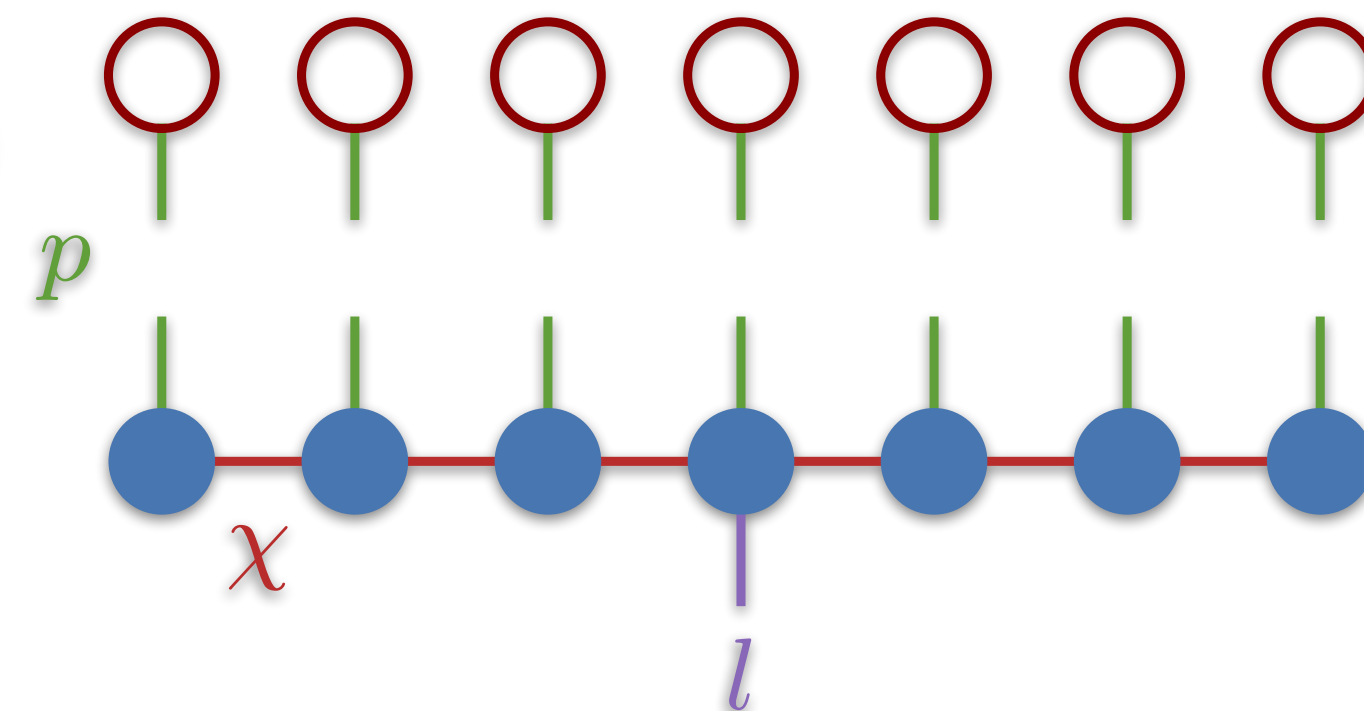
$$\phi^{p_i}(x_i) = \begin{bmatrix} \cos(x_i \pi/2) \\ \sin(x_i \pi/2) \end{bmatrix} \text{ or } \phi^{p_i}(x_i) = \begin{bmatrix} 1 \\ x_i \\ x_i^2 \end{bmatrix} \text{ or } \dots$$



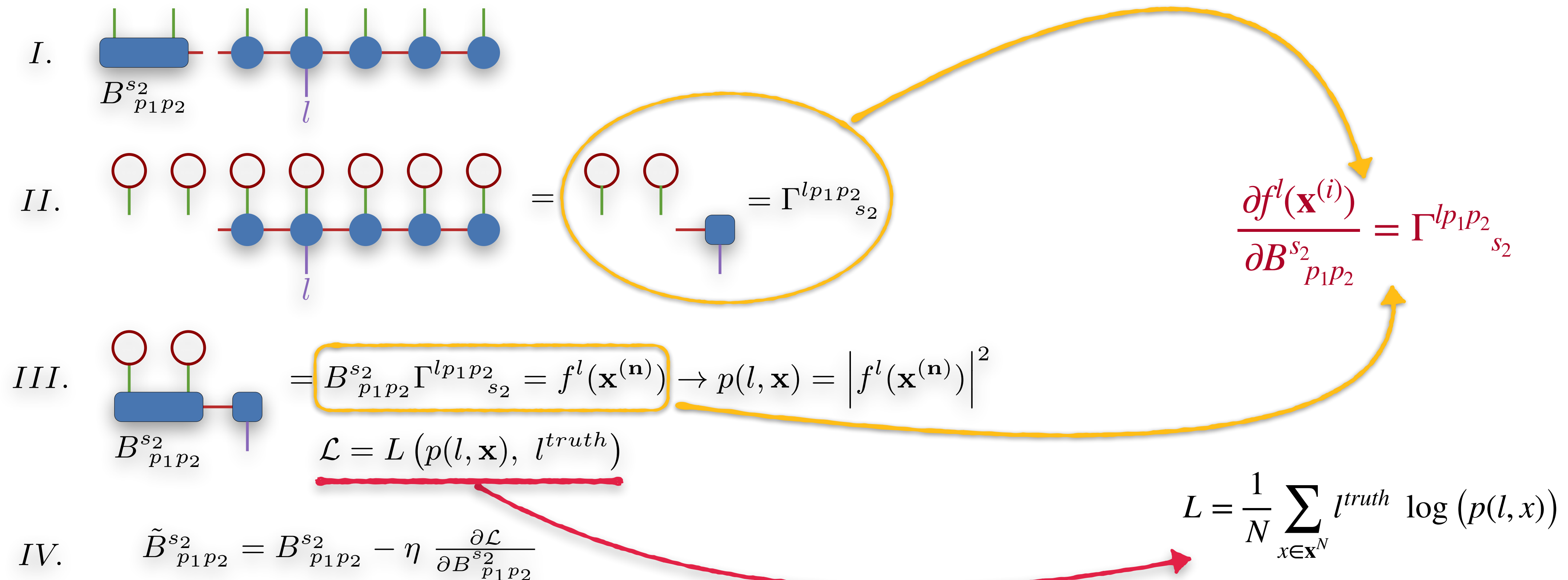
$$\text{[Seven red circles]} = \Phi^{p_1 \dots p_n}(\mathbf{x})$$



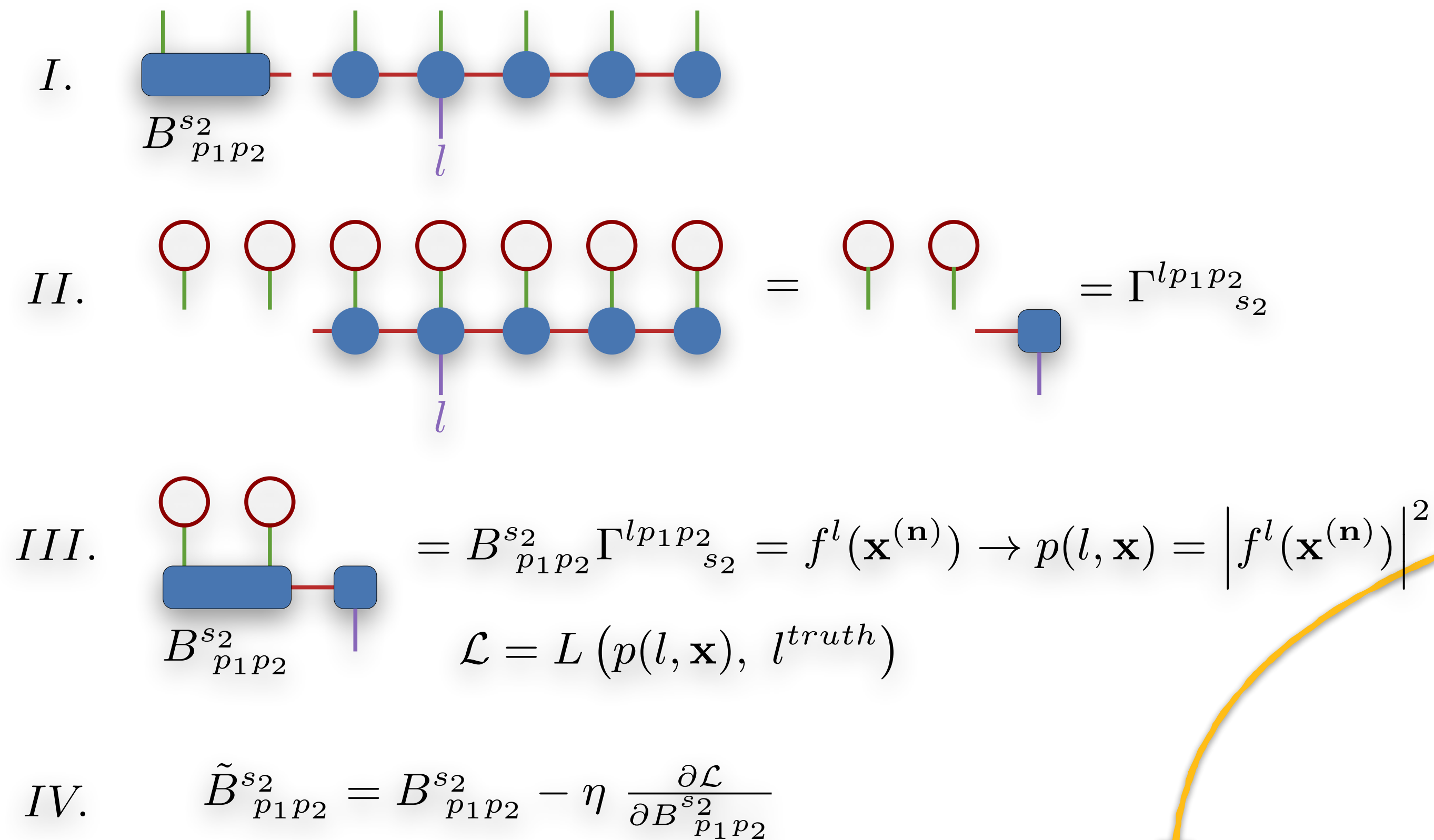
An efficient contraction algorithm is essential for training!



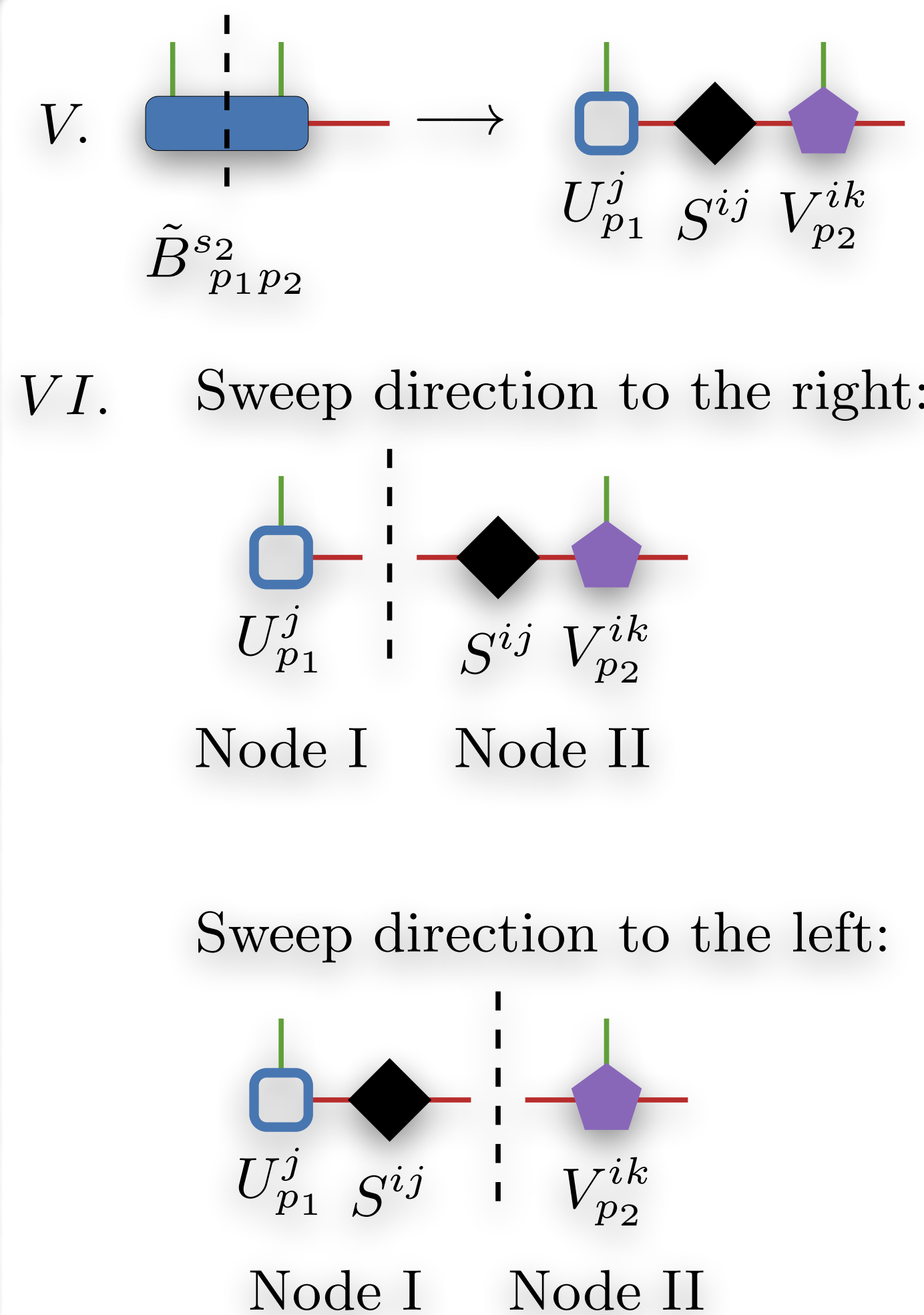
Density Matrix Renormalization Group Algorithm



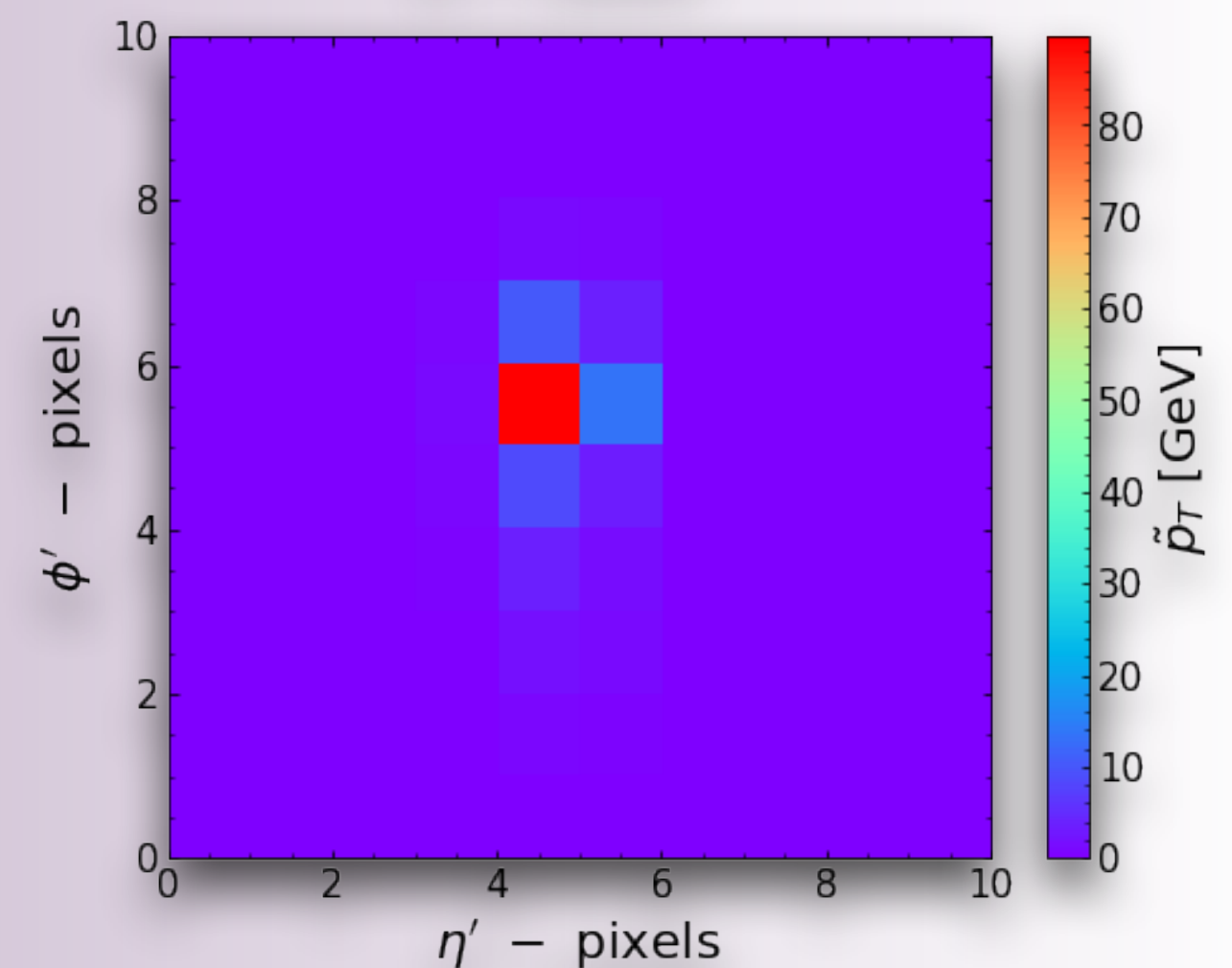
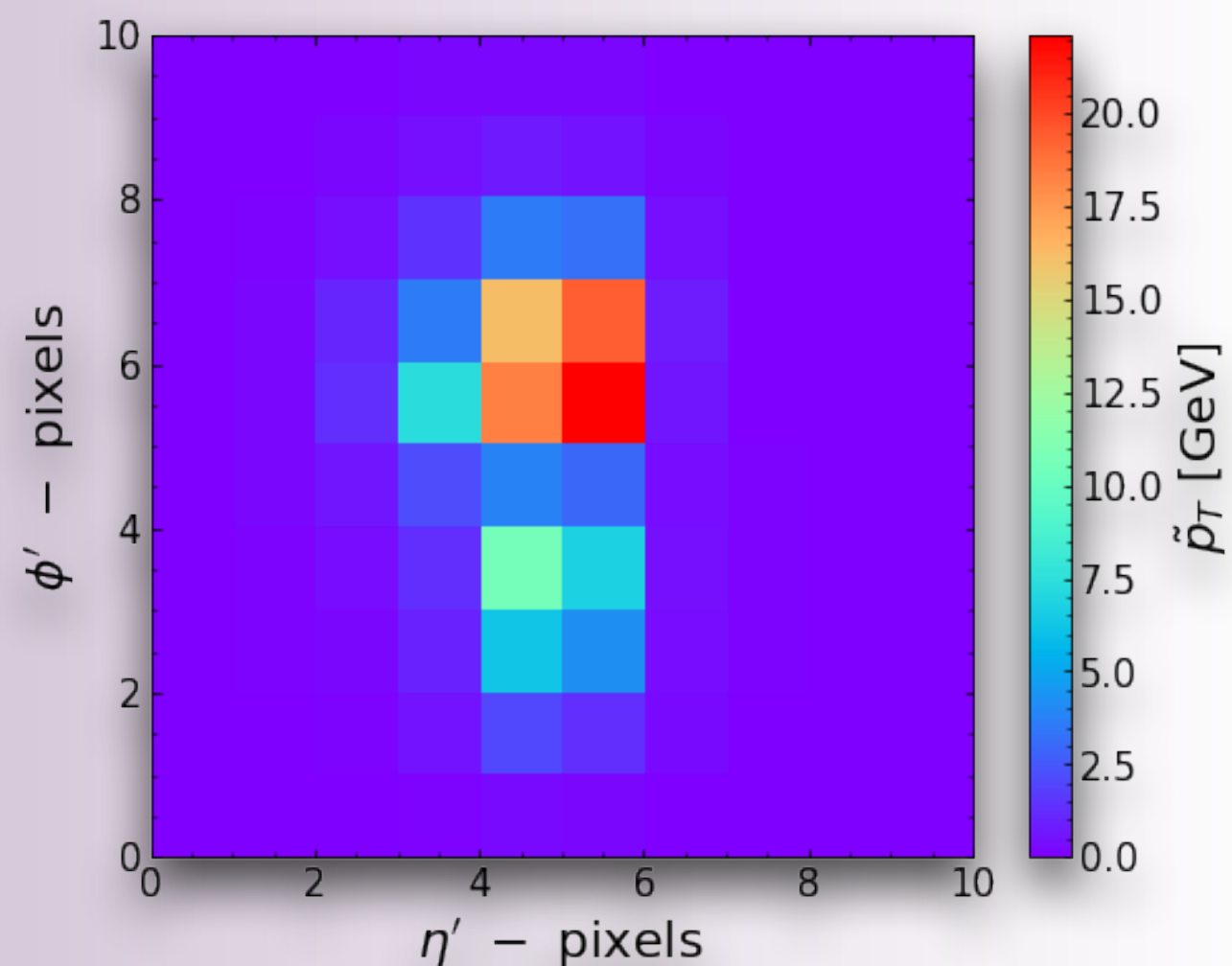
Density Matrix Renormalization Group Algorithm



Adjust the bond dimension via SVD!

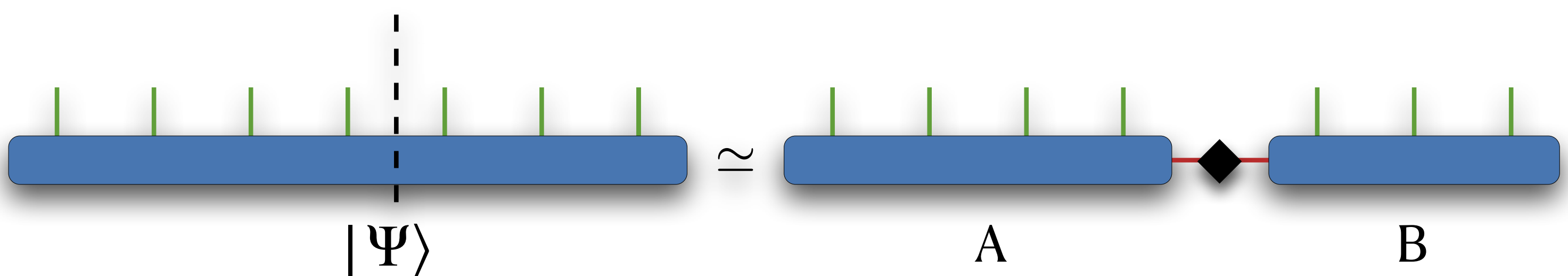


Top Tagging through MPS



Assumptions & Requirements

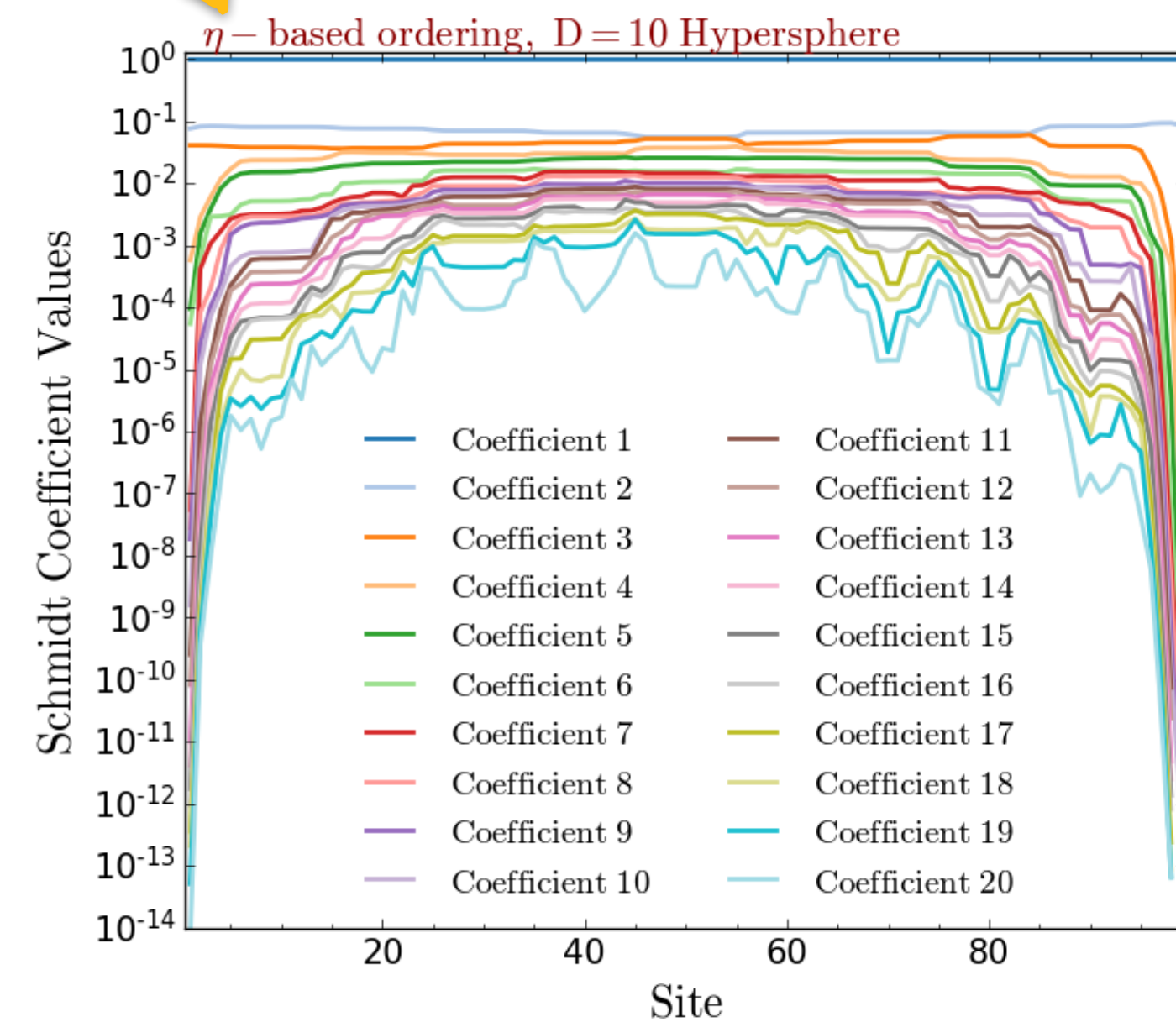
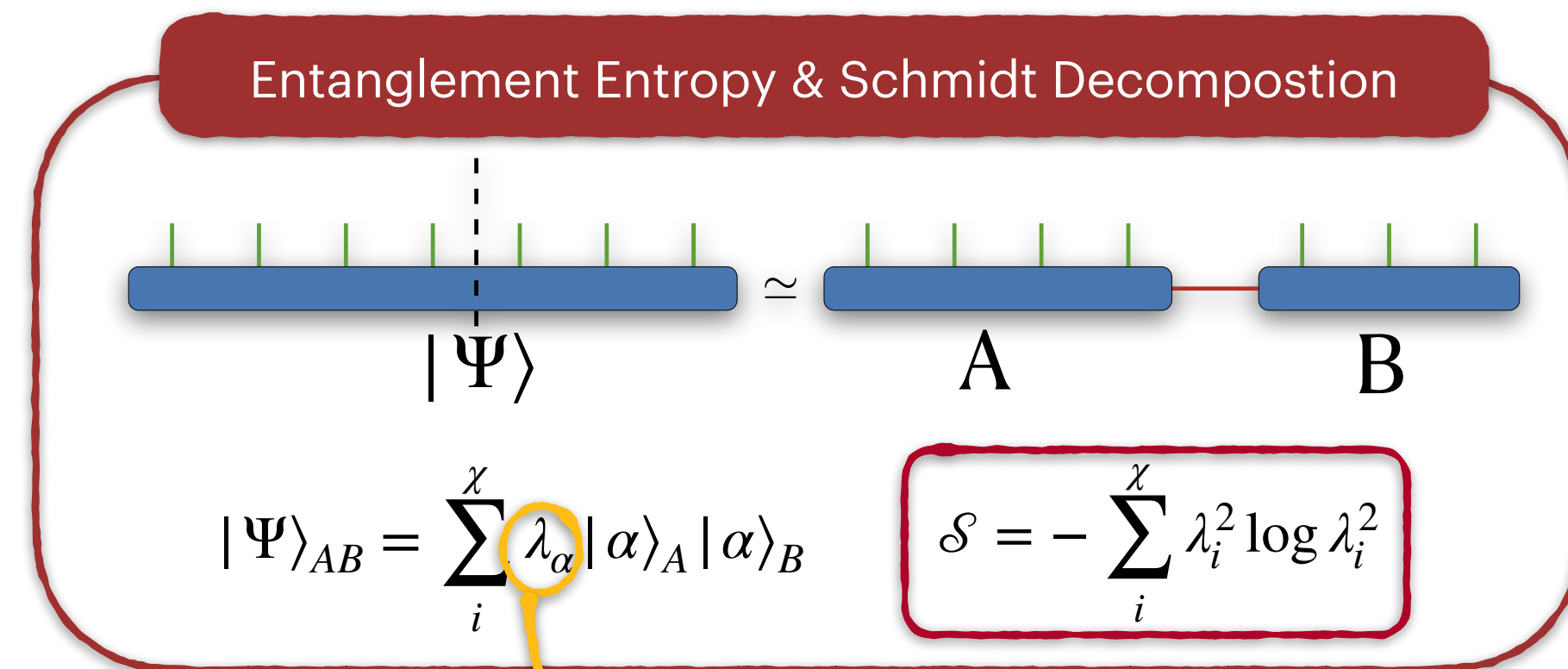
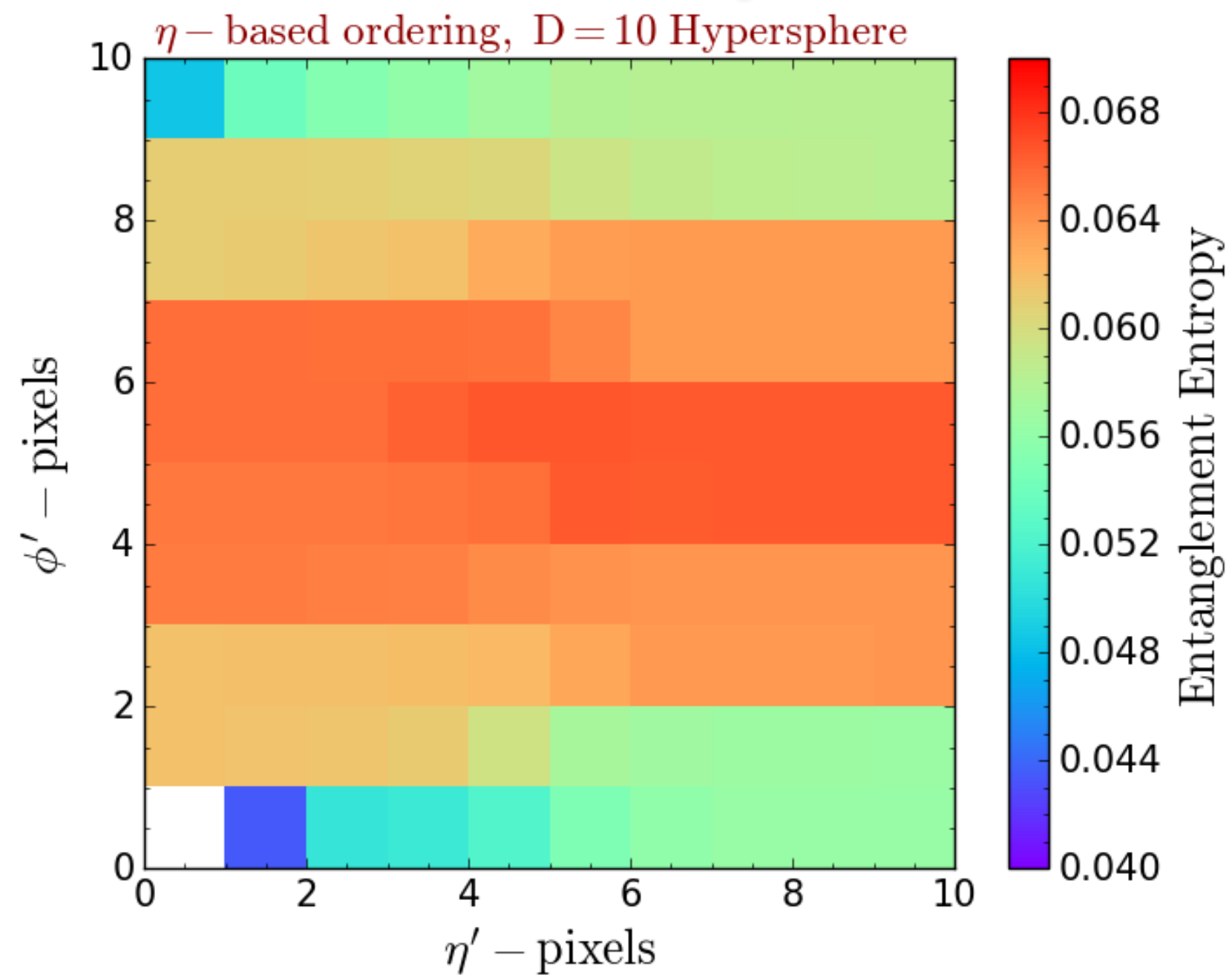
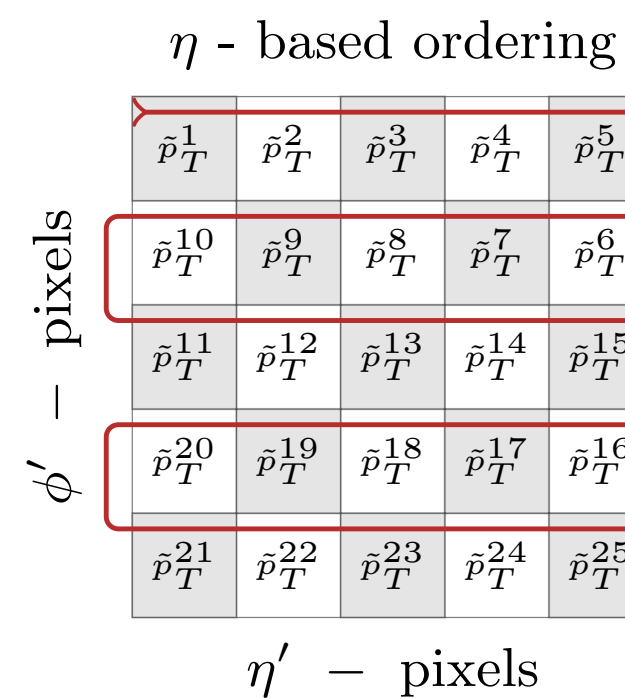
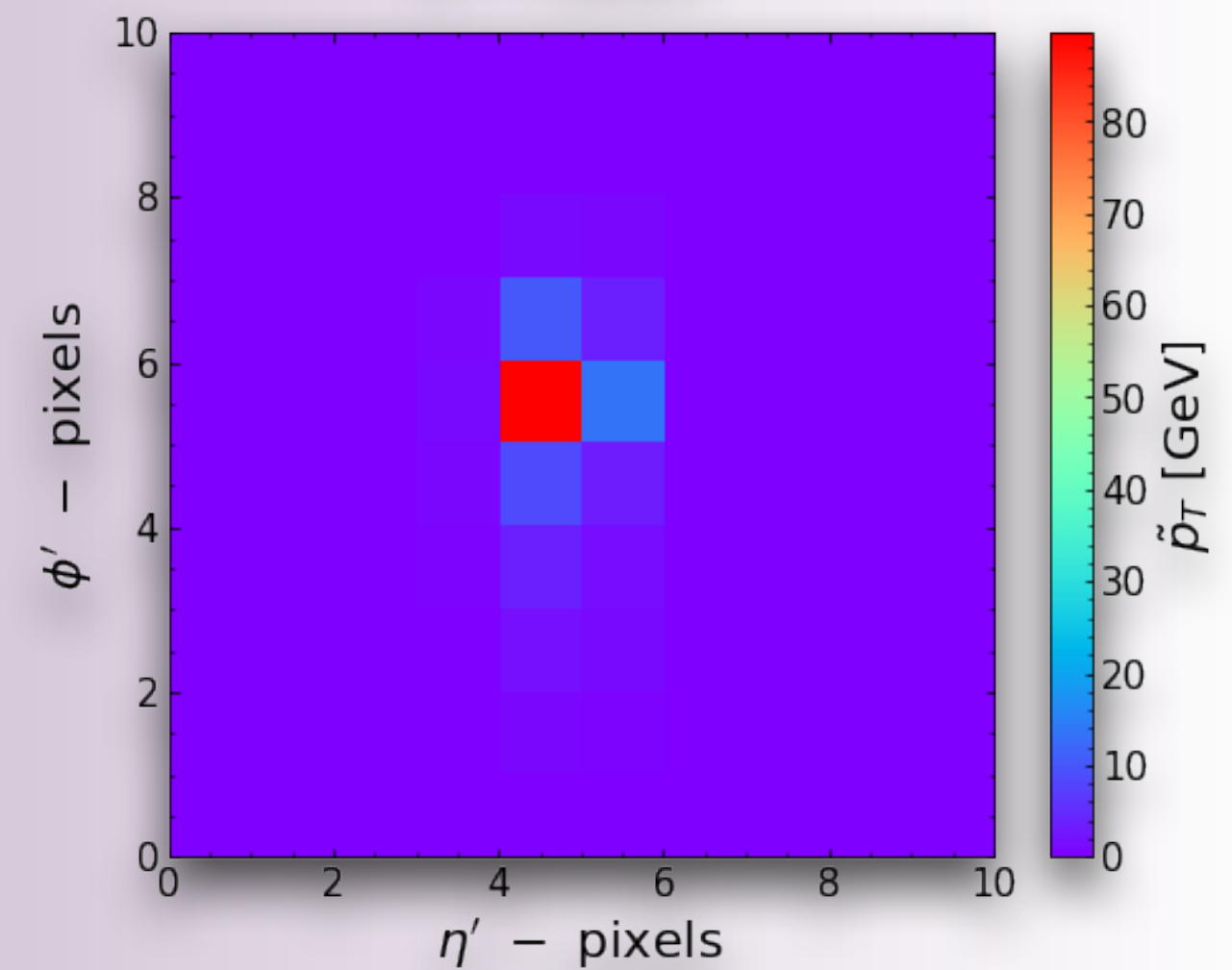
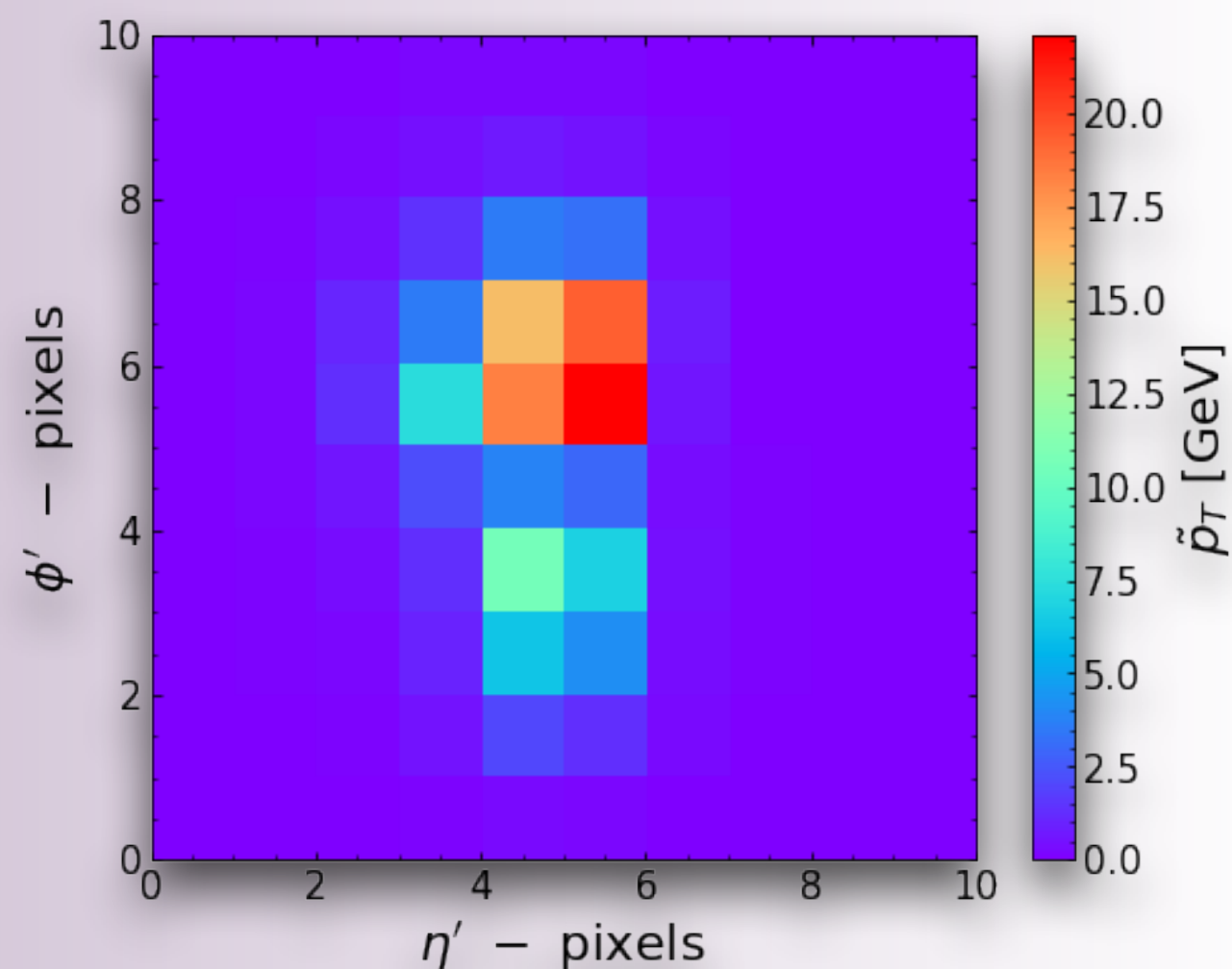
Entanglement Entropy & Schmidt Decomposition



$$|\Psi\rangle_{AB} = \sum_i^{\chi} \lambda_{\alpha} |\alpha\rangle_A |\alpha\rangle_B \quad \rightarrow \quad \lambda_{\alpha} := \text{Schmidt values}$$

$$\mathcal{S} = - \sum_i^{\chi} \lambda_i^2 \log \lambda_i^2 \quad \rightarrow \quad \text{von Neumann entropy}$$

Top Tagging through MPS



Fisher Information & Effective Dimensions

