

# Quantum States from Normalizing Flows

---

**Yukari Yamauchi**

*Based on:* arXiv:2406.02451 with **Scott Lawrence** and **Arlee Shelby**

March 19th, 2025, Stony Brook University

*Probing the frontiers of nuclear physics with AI at the EIC (II)*



# Neural network quantum states

## Solving the Quantum Many-Body Problem with Artificial Neural Networks

Giuseppe Carleo<sup>✉</sup>

*Theoretical Physics, ETH Zurich, 8093 Zurich, Switzerland*

Matthias Troyer

*Theoretical Physics, ETH Zurich, 8093 Zurich, Switzerland*

*Quantum Architectures and Computation Group,*

*Microsoft Research, Redmond, WA 98052, USA and*

*Station Q, Microsoft Research, Santa Barbara, CA 93106-6105, USA*

For a given quantum system

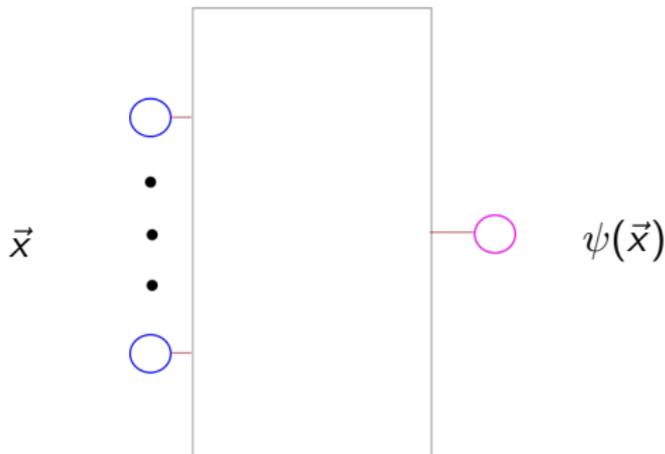
Represent a restricted Hilbert space with a neural network

Why neural network?

- Flexibility
- Efficiency

Problems to tackle

- ground state
- time evolution



# Applications in nuclear physics: ground states

Variational Monte Carlo calculations of  $A \leq 4$  nuclei with an artificial neural-network correlator ansatz

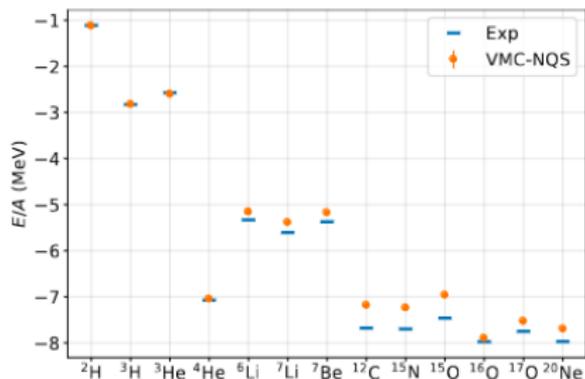
Corey Adams,<sup>1,2</sup> Giuseppe Carleo,<sup>3</sup> Alessandro Lovato,<sup>1,4</sup> and Noemi Rocco<sup>4,5</sup>

Hidden-nucleons neural-network quantum states for the nuclear many-body problem

Alessandro Lovato,<sup>1,2</sup> Corey Adams,<sup>1,3</sup> Giuseppe Carleo,<sup>4</sup> and Noemi Rocco<sup>5</sup>

Distilling the essential elements of nuclear binding via neural-network quantum states

Alex Gnech,<sup>1,2</sup> Bryce Fore,<sup>3</sup> and Alessandro Lovato<sup>2,3,4</sup>



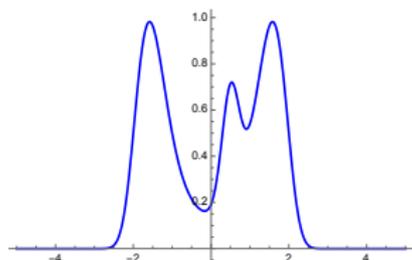
(A. Gnech, B. Fore, and A. Lovato, arXiv:2308.16266 [nucl-th])

# Normalizing flows

Normalizing flow  $\vec{x} = f(\vec{y})$ :

$$\det \left( \frac{\partial \vec{y}}{\partial \vec{x}} \right) p(\vec{x}) = \mathcal{N} \prod_{i=1}^N \frac{1}{\sqrt{2\pi}} e^{-y_i^2/2}$$

Some distribution

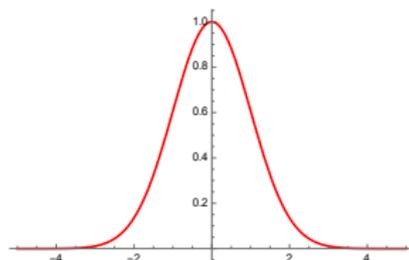


Map

$\leftrightarrow$

$x(y)$

Gaussian distribution



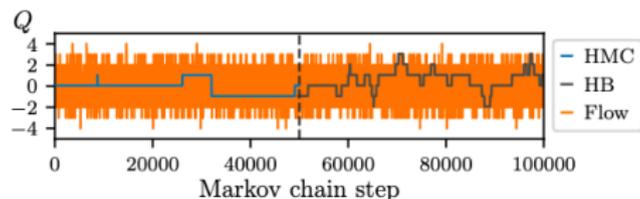
if such a map exists, then

$$\langle \mathcal{O} \rangle = \frac{\int d\vec{x} p(\vec{x}) \mathcal{O}(\vec{x})}{\int d\vec{x} p(\vec{x})} = \frac{\int d\vec{y} G_N(\vec{y}) \mathcal{O}(\vec{x}(\vec{y}))}{\int d\vec{y} G_N(\vec{y})}$$

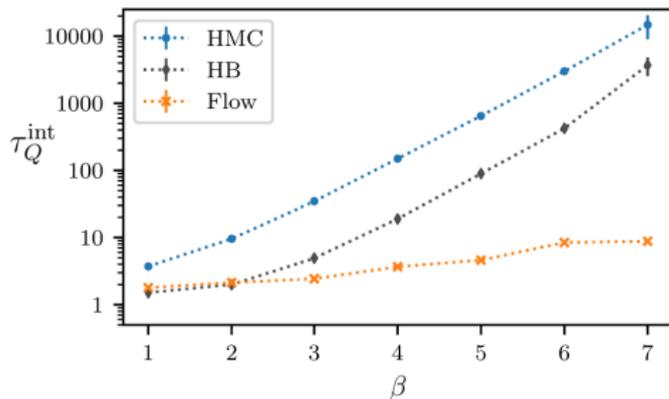
# Normalizing flows for lattice path integrals

Path integral of lattice  $U(1)$  gauge theory<sup>1</sup>  $Z = \int [dU] e^{-S(U)}$

$[dU] e^{-S(U)} \leftrightarrow$  decoupled uniform distribution on circles



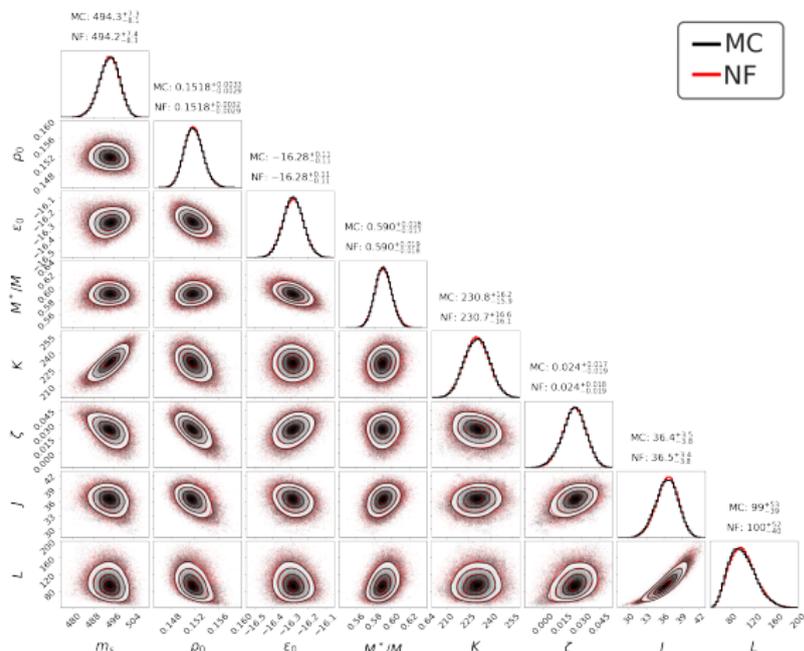
The autocorrelation of samples decreased dramatically with **Flow**



<sup>1</sup>G. Kanwar, M.S. Alberg, D. Boyda, K. Cranmer, D. Hackett, S. Racanière, D. Jimenez Rezende, and P.E. Shanahan.  
arXiv:2003.06413

# Normalizing flows for Bayesian posteriors<sup>2</sup>

## A relativistic mean field model



- $m_s$ :  $\sigma$  meson mass
- $\rho_0$ : saturation density
- $\epsilon_0$ : binding energy at  $\rho_0$
- $M^*$ : effective nucleon mass at  $\rho_0$
- $K$ : incompressibility at  $\rho_0$
- $J$ : value of symmetric energy at  $\rho_0$
- $L$ : slope of symmetric energy at  $\rho_0$
- $\zeta$ :  $\omega$  meson quartic coupling

All codes developed in this work are available at [https://gitlab.com/yyamauchi/uq\\_nf](https://gitlab.com/yyamauchi/uq_nf)

<sup>2</sup>YY, L. Buskirk, P. Giuliani, and K. Godbey, arXiv:2310.04635 [nucl-th]

# Normalizing flows for quantum states?

1. **Ground states via QuantumNVP (from RealNVP)**
2. **Time evolution via quantum continuous normalizing flows**

# RealNVP neural network

RealNVP<sup>3</sup> is a sequence of Affine coupling layers

$$\vec{x} = A(\vec{y}; \alpha) = \vec{b}_\alpha \odot \vec{y} + (I - \vec{b}_\alpha) \odot \left[ \vec{y} \odot \exp \left[ s(\vec{b}_\alpha \odot y) \right] + t(\vec{b}_\alpha \odot y) \right]$$

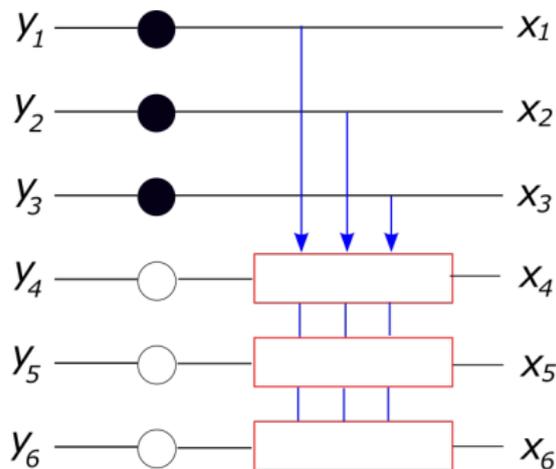
Network parameters:

- masking  $\vec{b}_\alpha$ :  $[1, 0, 1, 0, \dots]$  or  $[0, 1, 0, 1, \dots]$
- scaling  $s$ : 1-layer MLP
- translation  $t$ : Linear

Benefits:

- Easily invertible
- Simple  $\det J$

$$\det \left( \frac{\partial x}{\partial y} \right) = \prod_{i=1, \dots, N_p} \exp(s_i(1 - b_i))$$



<sup>3</sup>L. Dinh, J. Sohl-Dickstein, and S. Bengio. arXiv:1605.08803

# QuantumNVP

$$\vec{x} = Q(\vec{y}; \alpha) = \vec{b}_\alpha \odot \vec{y} + (I - \vec{b}_\alpha) \odot \left[ \vec{y} \odot |s(\vec{b}_\alpha \odot \vec{y})|^2 + t(\vec{b}_\alpha \odot \vec{y}) \right]$$

**Scaling  $s$  is complexified!!**

The entire flow is

$$\begin{aligned} \vec{x} &= Q_{N_l} \circ Q_{N_l-1} \cdots \circ Q_2 \circ Q_1(\vec{y}) \\ \det \left( \frac{\partial \vec{x}}{\partial \vec{y}} \right) &= \prod_{j=1}^{N_l} \prod_{i=1}^{N_p} |qs_j|_i^{2(1-b_i)} \end{aligned}$$

The wave function is

$$\psi(\vec{x}) = \prod_{j=1}^{N_l} \prod_{i=1}^{N_p} (qs_j)_i^{1-b_i}, \quad \prod_{i=1}^{N_p} \frac{1}{(2\pi)^{1/4}} \exp(-y_i^2/4)$$

In  $y$  space,

$$d\vec{x} \psi^\dagger(\vec{x})\psi(\vec{x}) = \prod_{i=1}^{N_p} dy_i \frac{1}{(2\pi)^{1/2}} \exp(-y_i^2/2)$$

# Finding the ground state

Minimization problem within the restricted Hilbert space

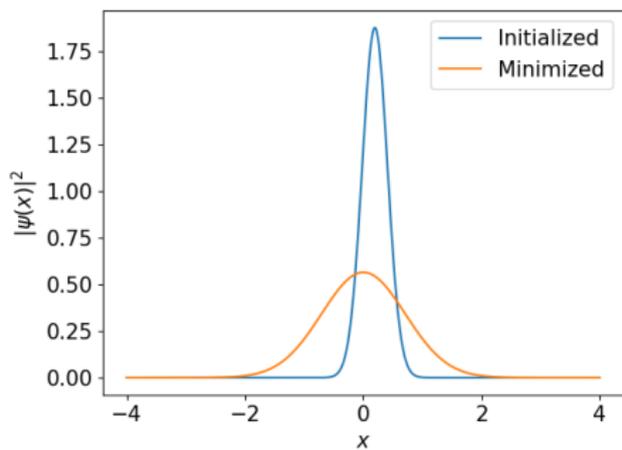
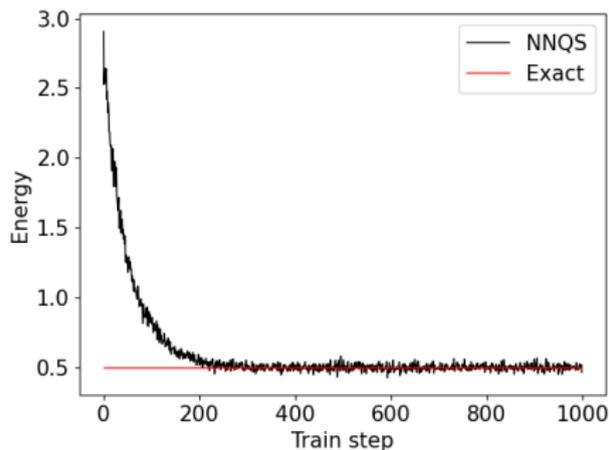
$$E_0 = \min (\langle \psi | H | \psi \rangle)$$

In quantum mechanics  $H = -\frac{\nabla^2}{2} + V(x)$ , so

$$\langle H \rangle = \int dx \frac{1}{2} \frac{d\psi^*(x)}{dx} \frac{d\psi(x)}{dx} + \psi^*(x) V(x) \psi(x)$$

Example: Harmonic oscillator

$$H = -\frac{\nabla^2}{2} + \frac{x^2}{2}$$

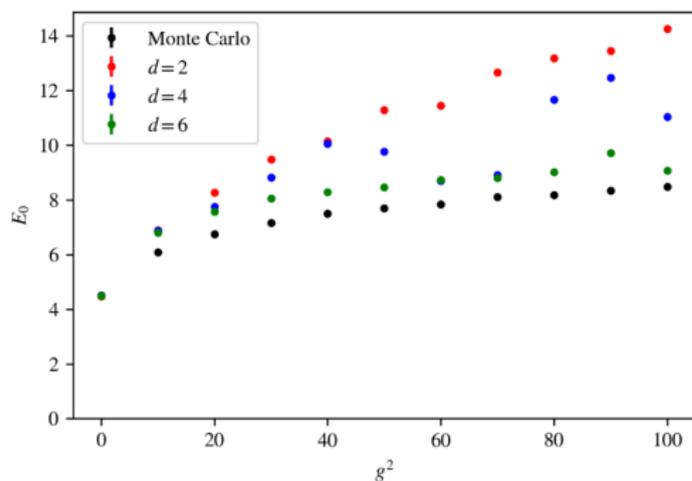


# Three particles in three spatial dimensions

Intecacting via Yukawa potential in an external harmonic trap

$$\hat{H}_{\text{trap}} = \sum_n \left( \frac{\hat{p}_n^2}{2M} + \frac{M\omega^2}{2} \hat{x}_n^2 \right) + \sum_{n < m} V_{nm}(\hat{x})$$

$$\text{where } V_{nm}(x) = \frac{g^2}{|x_n - x_m|} e^{-M|x_n - x_m|}$$



Demonstration:  $M = \omega = 1$

- **Adam** training with learning rate  $3 \times 10^{-4}$
- $2^{10}$  samples per train step
- $2^{15}$  samples to evaluate energy

Compared to Monte Carlo at  $\beta = 10$

# Time evolution in quantum mechanics

Two challenges:

- We cannot represent continuous time
- Exact  $\psi(x)$  will likely not be in QuantumNVP, thus error will pile up

Let us, first, discretize time

$$i [|\psi(t + \delta)\rangle - |\psi(t)\rangle] \approx \delta \hat{H} \frac{\psi(t + \delta) + \psi(t)}{2}$$

Now time evolution can be trained with

$$L[\psi(t + \delta), \psi(t)] = \int dx \left| \frac{\psi(t + \delta) - \psi(t)}{\delta} + i \frac{\hat{H}\psi(t + \delta) + \hat{H}\psi(t)}{2} \right|^2$$

**The violation of the loss function gives uncertainty**

# Time evolution via NFQS

## Simulation hyperparameters

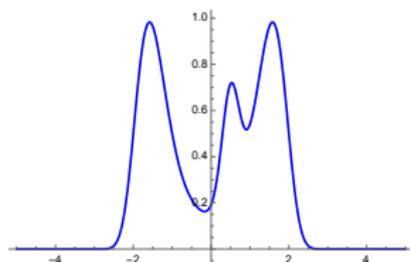
- Time step  $\delta$
- Threshold  $t$  on the loss function
- Training hyperparameters

Simulation needs 2 copies of wavefunctions:  $\psi, \psi'$

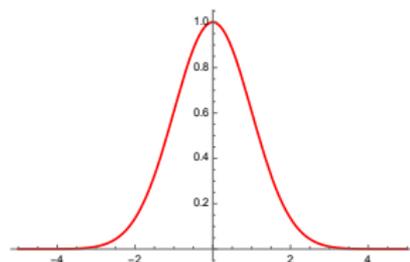
- 1 Prepare the initial state via NF
- 2 Perform one time step by
  - 1 Train  $\psi'$  for  $\delta$  time ahead
  - 2 Copy  $\psi'$  to  $\psi$
  - 3 Save  $\psi$  as the wavefunction at the time
- 3 Repeat

# Continuous normalizing flows

Some distribution



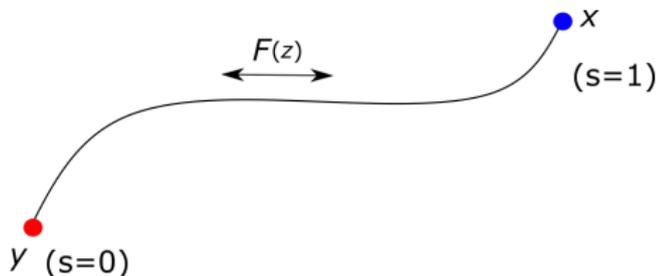
Gaussian distribution



Map  
 $\leftrightarrow$   
 $x(y)$

Map  $x \leftrightarrow y$  via  $\frac{dz}{ds} = F(z; \sigma)$

- $F(z)$ : 3-layer MLP ( $\tanh(y)$ )
- Invertible



# Quantum continuous normalizing flows

Wavefunction is complex-valued

**Can't be generated by standard normalizing flow**

The neural network

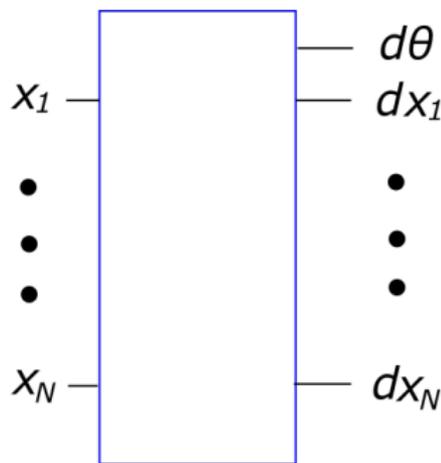
$$\vec{x} \rightarrow \left( \frac{d\vec{x}}{ds}, \frac{d\theta}{ds} \right)$$

The wavefunction

$$\psi(\vec{x}) = \sqrt{\det \left( \frac{\partial \vec{y}}{\partial \vec{x}} \right)} e^{i\theta(\vec{x})} \prod_i \frac{1}{(2\pi)^{1/4}} e^{-y_i(\vec{x})^2/4}$$

In  $y$  space,

$$d\vec{x} \psi^\dagger(\vec{x}) \psi(\vec{x}) = \prod_i dy_i \frac{1}{(2\pi)^{1/2}} e^{-y_i^2/2}$$



**The wavefunction is always normalized to 1**

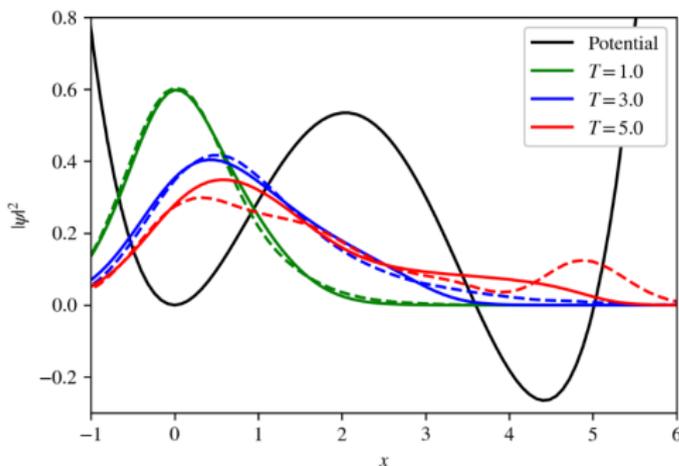
# Quantum tunneling in one spatial dimension

With the potential

$$H = \frac{\hat{p}^2}{2} + \frac{1}{2b^2}x^2(x-b)^2 - \frac{a}{b^3}, \quad a = 0.25, b = 0.45$$

- The **false vacuum** is with  $V = \frac{x^2}{2}$ .
- at  $x \approx 2.0$ , maximal potential is  $V \approx 0.55$ .

With  $\psi(t=0) = \pi^{-1/4}e^{-x^2/2}$ ,



# Uncertainty estimation from loss values

Overlap measure

$$E[\psi](t) = 1 - \text{Re}\langle\psi_{\text{true}}|\psi(t)\rangle = \frac{|\epsilon\rangle^2}{2}, \text{ with } |\epsilon\rangle = |\psi_{\text{true}}\rangle - |\psi(t)\rangle$$

The total error at time  $t$  can be estimated as

$$E[\psi](t) = \int_0^t dt' \frac{d}{dt'} E[\psi](t')$$

The derivative can be bounded as

$$\frac{d}{dt} E[\psi](t) = \text{Re}\langle\psi_{\text{true}}| \left[ iH + \frac{d}{dt} \right] |\psi\rangle \leq [\langle\chi|\chi\rangle]^{1/2}, \text{ where } \left[ iH + \frac{d}{dt} \right] |\psi\rangle$$

This is directly connected to loss function, therefore

$$E[\psi](t) = \int_0^t dt' \sqrt{L(t')}$$

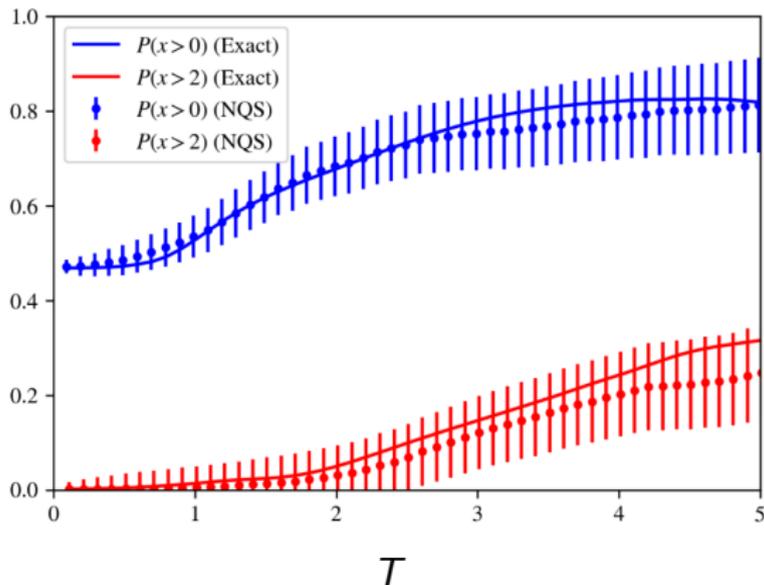
# Error analysis of amplitudes

For other operators, bounds can be similarly obtained:

$$\left| \langle \psi_{\text{true}} | \hat{\mathcal{O}} | \psi_{\text{true}} \rangle - \langle \psi | \hat{\mathcal{O}} | \psi \rangle \right| < \| \mathcal{O} \| \left( 2|\epsilon| + |\epsilon|^2 \right)$$

For example,

$$\hat{\mathcal{O}} = \theta(x - x_0)$$



**The error is integrated over as a random walk**

# Future

- Symmetric / antisymmetric wavefunctions
- Applications to low-energy nuclear reactions
- Applications to neutrino many-body problems
- Any applications in EIC physics?

**Thank you!**