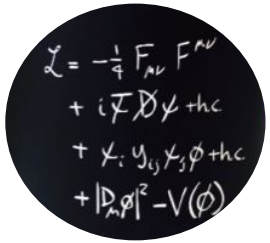
Build Big or Build Smart: Examining Scale and Domain Knowledge in Machine Learning for Fundamental Physics

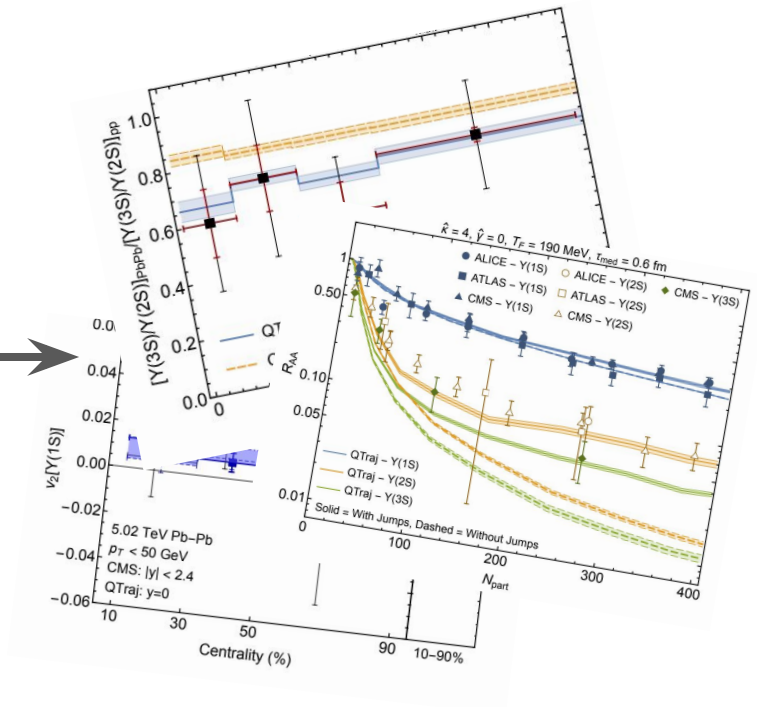
Neural network quantum states and simulation of real-time dynamics of quantum systems

Tom Magorsch
Technical University of Munich

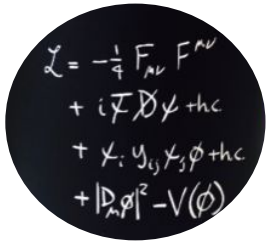
28.08.2025

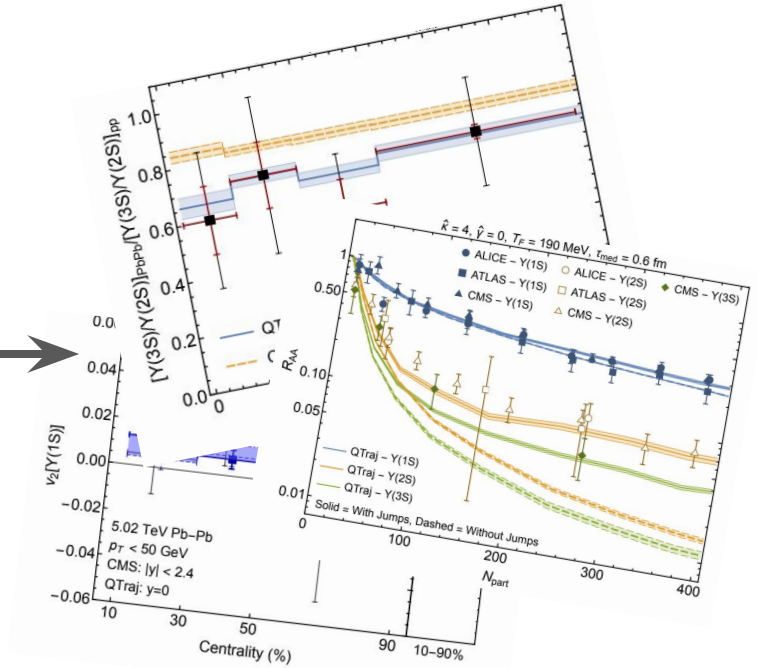
Predictions from the Standard model





Predictions from the Standard model



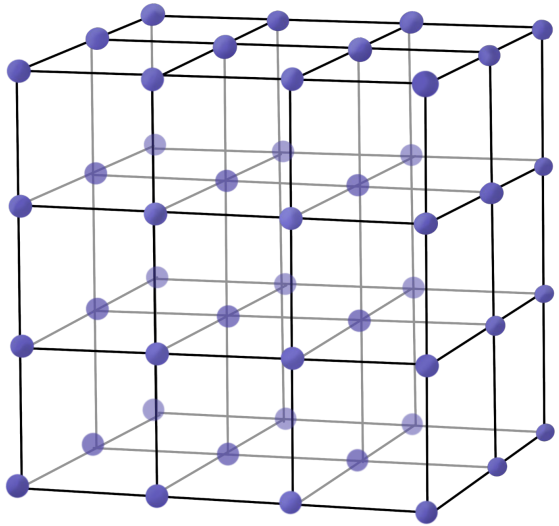


Lattice QCD

$$\int \mathcal{D}\phi \, e^{iS[\phi]}$$

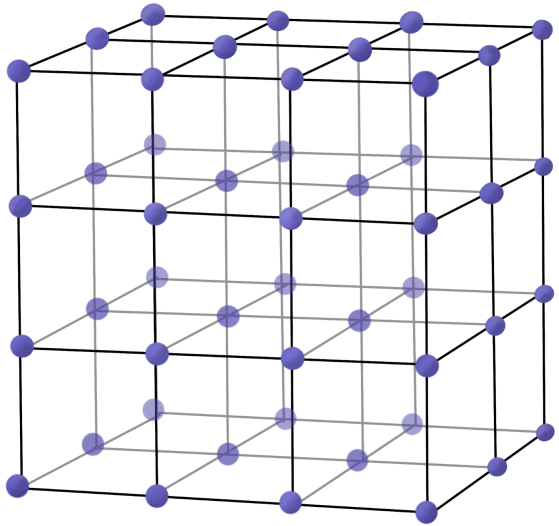
Lattice QCD

$$\int \mathcal{D}\phi \, e^{iS[\phi]}$$



Lattice QCD

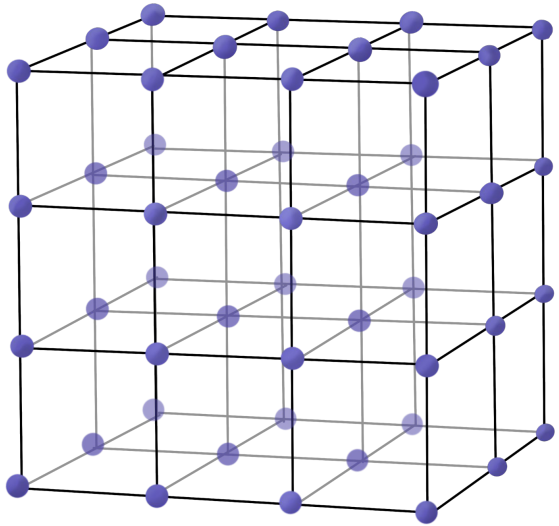
$$\int \prod_x d\phi_x \, e^{iS[\phi]}$$



Lattice QCD

$$\int \prod_x d\phi_x \, e^{iS[\phi]}$$

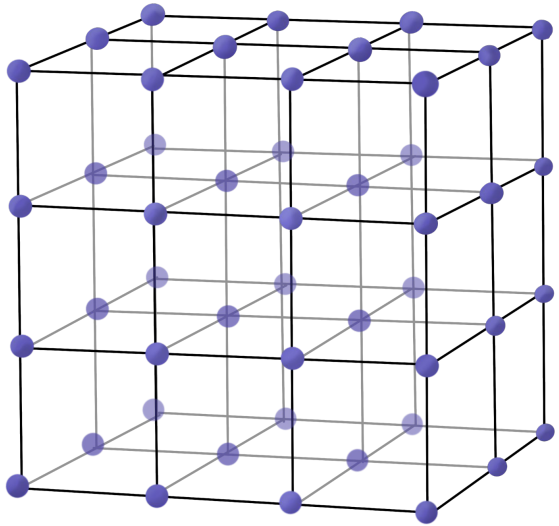
$$24^3 \times 64 \approx 900\text{k}$$



Lattice QCD

$$t \to -i\tau$$

$$\int \prod_x d\phi_x \, e^{-S_E[\phi]}$$

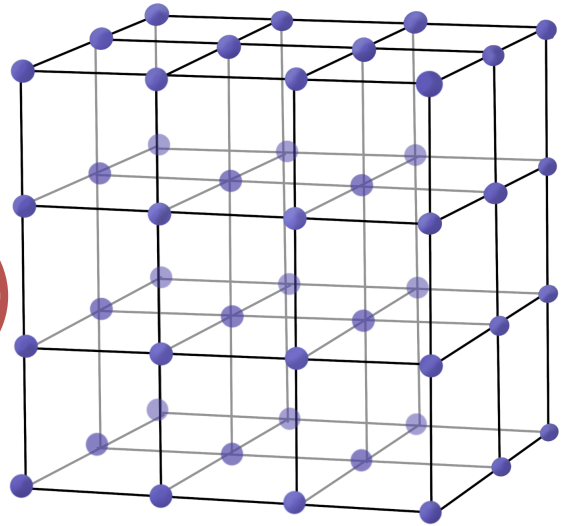


Lattice QCD

$$t \to -i\tau$$

$$\int \prod_x d\phi_x \, e^{-S_E[\phi]}$$

$$p(\phi)$$



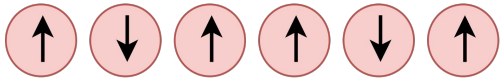
Hamiltonian methods

$$|\phi\rangle \in \mathcal{H}$$

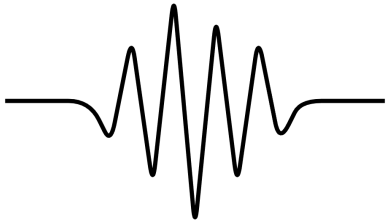
$$i\frac{d}{dt}|\phi(t)\rangle = H|\phi(t)\rangle$$

Simulating large Quantum systems

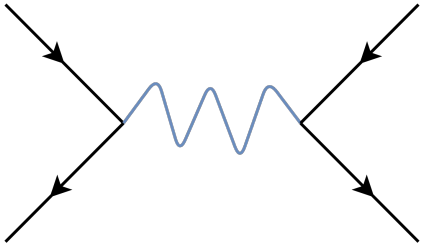
Spin Systems



Non-Rel. Systems



QFTs



Example: Ground state energy

$$E = \frac{\langle \psi | H | \psi \rangle}{\langle \psi | \psi \rangle}$$

Example: Ground state energy

$$E = \frac{\langle\psi|H|\psi\rangle}{\langle\psi|\psi\rangle}$$

$$H|\psi\rangle = E|\psi\rangle$$

Diagonalize H

Example: Ground state energy

$$E = \frac{\langle \psi_\Theta | H | \psi_\Theta \rangle}{\langle \psi_\Theta | \psi_\Theta \rangle}$$

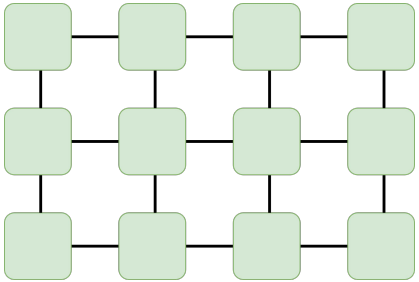
$$\Theta \leftarrow \Theta - \alpha \nabla_\Theta E$$

Variational states

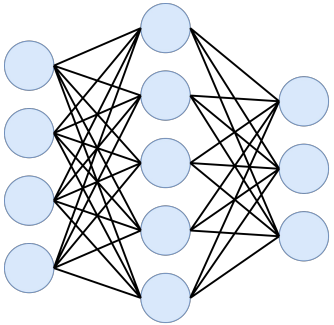
Hartree–Fock

$$\psi_\Theta \propto \begin{vmatrix} \phi_{\Theta_1} & \phi_{\Theta_2} & \cdots \\ \phi_{\Theta_3} & \phi_{\Theta_4} & \cdots \\ \vdots & \vdots & \ddots \end{vmatrix}$$

Tensor Networks



Neural Network



Expressiveness

Variational Monte Carlo

$$E = \frac{\langle \psi_\Theta | H | \psi_\Theta \rangle}{\langle \psi_\Theta | \psi_\Theta \rangle}$$

Variational Monte Carlo

$$E = \mathbb{E}_{p(x)} \left[\frac{\langle x | H | \psi_\Theta \rangle}{\langle x | \psi_\Theta \rangle} \right]$$

$$p(x) = \frac{|\langle x | \psi_\Theta \rangle|^2}{\sum_x |\langle x | \psi_\Theta \rangle|^2}$$

Variational Monte Carlo

$$E = \mathbb{E}_{p(x)} \left[\overbrace{\frac{\langle x | H | \psi_\Theta \rangle}{\langle x | \psi_\Theta \rangle}}^{E_{\mathrm{loc}}(x)} \right]$$

$$p(x) = \frac{|\langle x | \psi_\Theta \rangle|^2}{\sum_x |\langle x | \psi_\Theta \rangle|^2}$$

Variational Monte Carlo

$$\begin{pmatrix} 0 & 1 & 0 & \dots \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & \dots \\ 1 & 0 & 1 & \dots \\ 0 & 1 & 0 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \begin{pmatrix} \psi_\Theta(x_1) \\ \psi_\Theta(x_2) \\ \psi_\Theta(x_3) \\ \vdots \end{pmatrix}$$

$$E_{\mathrm{loc}}(x) = \frac{\langle x | H | \psi_\Theta \rangle}{\langle x | \psi_\Theta \rangle}$$

Variational Monte Carlo

$$\begin{pmatrix} 0 & 1 & 0 & \dots \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & \dots \\ 1 & 0 & 1 & \dots \\ 0 & 1 & 0 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \begin{pmatrix} \psi_\Theta(x_1) \\ \psi_\Theta(x_2) \\ \psi_\Theta(x_3) \\ \vdots \end{pmatrix}$$

$$= \psi_\Theta(x_1) + \psi_\Theta(x_3)$$

Variational Monte Carlo

$$\nabla_\Theta \langle H \rangle = \nabla_\Theta \mathbb{E}_{p(x|\Theta)} \left[\frac{\langle x | H | \psi_\Theta \rangle}{\langle x | \psi_\Theta \rangle} \right]$$

Variational Monte Carlo

$$\nabla_\Theta \langle H \rangle = \nabla_\Theta \mathbb{E}_{p(x|\Theta)} \left[\frac{\langle x | H | \psi_\Theta \rangle}{\langle x | \psi_\Theta \rangle} \right]$$

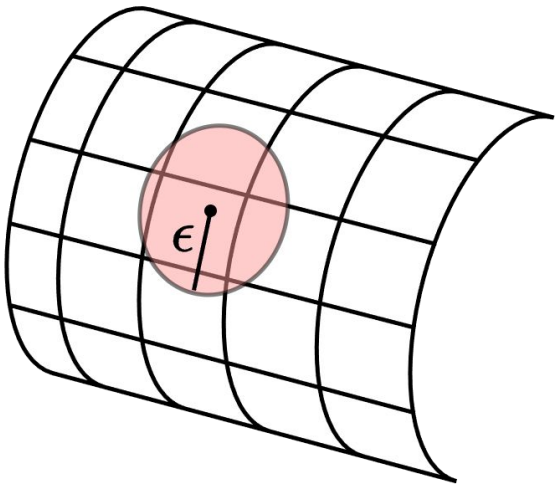
$$= 2\mathrm{Re}\, \mathbb{E}_{p(x|\Theta)} \left[\nabla_\Theta \log \psi_\Theta(x)^* \left(E_{\mathrm{loc}}(x, \Theta) - \langle E_{\mathrm{loc}} \rangle \right) \right]$$

Stochastic reconfiguration

Natural gradient descent:

$$F_{ij} = \mathbb{E}[\partial_i \log p_\Theta(x) \partial_j \log p_\Theta(x)]$$

$$\Theta \leftarrow \Theta - \alpha F^{-1} \nabla_\Theta L$$

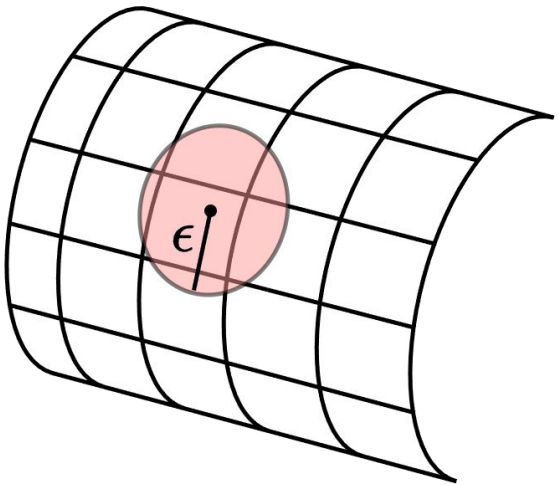


Stochastic reconfiguration

$$S_{ij} = \mathrm{Re}\left[\langle \partial_i \log \psi^* \partial_j \log \psi \rangle - \langle \partial_i \log \psi^* \rangle \langle \partial \log \psi \rangle\right]$$

$$g = 2\mathrm{Re}\,\mathbb{E}_{p(x|\Theta)}\left[\nabla_\Theta \log \psi_\Theta(x)^* \left(E_{\mathrm{loc}}(x,\Theta) - \langle E_{\mathrm{loc}} \rangle\right)\right]$$

$$\Theta \leftarrow \Theta - \alpha S^{-1} g$$



Time evolution: TDVP

$$\frac{d}{dt}|\psi_\Theta\rangle = -iH|\psi_\Theta\rangle$$

Time evolution: TDVP

$$\frac{d}{dt}|\psi_\Theta\rangle = -iH|\psi_\Theta\rangle$$

$$\frac{d}{dt}|\psi_\Theta\rangle = \sum_i \frac{d\Theta_i}{dt}\frac{d}{d\Theta_i}|\psi_\Theta\rangle$$

Time evolution: TDVP

Variational principle:

$$\min \left\| \sum_i \frac{d\Theta_i}{dt} \frac{d}{d\Theta_i} |\psi_\Theta\rangle - iH|\psi_\Theta\rangle \right\|^2$$

Time evolution: TDVP

$$S_{ij} = \mathrm{Re}\left[\langle \partial_i \log \psi^* \partial_j \log \psi \rangle - \langle \partial_i \log \psi^* \rangle \langle \partial \log \psi \rangle\right]$$

$$F_i = 2\mathrm{Re}\left[\langle \partial_i \log \psi^* E_{\mathrm{loc}} \rangle - \langle \partial_i \log \psi^* \rangle \langle E_{\mathrm{loc}} \rangle\right]$$

$$\frac{d\Theta}{dt} = S^{-1} F$$

solve time evolution of the parameters with an ODE integrator!

Time evolution: TDVP

$$S_{ij} = \mathrm{Re}\left[\langle \partial_i \log \psi^* \partial_j \log \psi \rangle - \langle \partial_i \log \psi^* \rangle \langle \partial \log \psi \rangle\right]$$

$$F_i = 2\mathrm{Re}\left[\langle \partial_i \log \psi^* E_{\mathrm{loc}} \rangle - \langle \partial_i \log \psi^* \rangle \langle E_{\mathrm{loc}} \rangle\right]$$

$$\frac{d\Theta}{dt} = S^{-1} F$$

solve time evolution of the parameters with an ODE integrator!

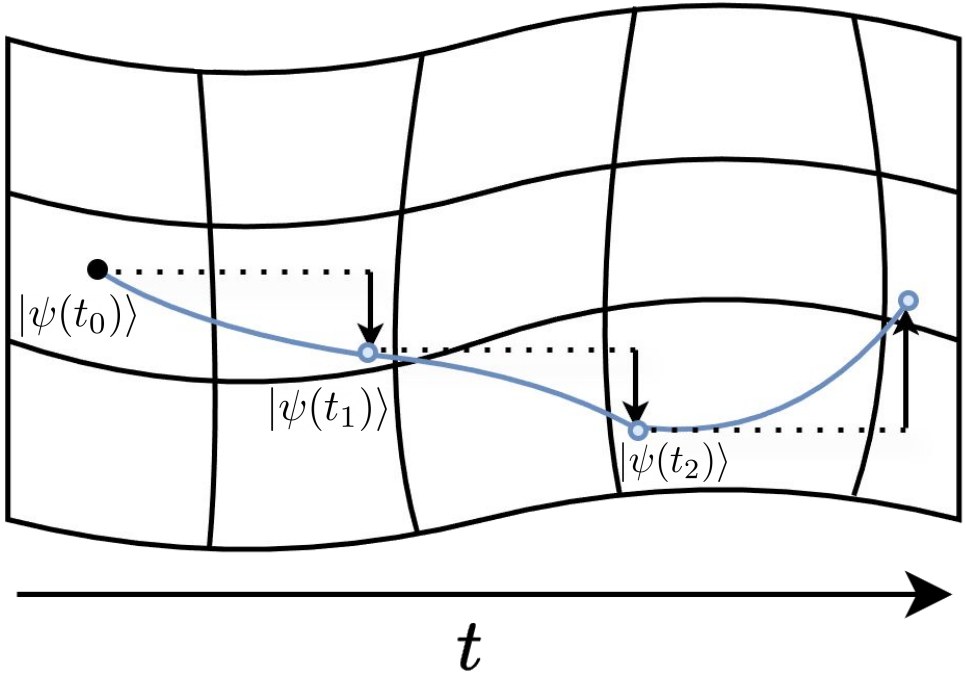
Potential issues:
- Spectrum of S (regularize)
- Expensive for large models

Sinibaldi, A., Giuliani, C., Carleo, G., & Vicentini, F. (2023). Unbiasing time-dependent Variational Monte Carlo by projected quantum evolution. *Quantum*, *7*, 1131.

Time evolution: p-tVMC

Minimize **Infidelity:**

$$I = 1 - \frac{|\langle\psi(t_1)|\psi(t_0)\rangle|^2}{\langle\psi(t_1)|\psi(t_1)\rangle\langle\psi(t_0)|\psi(t_1)\rangle}$$



Sinibaldi, A., Giuliani, C., Carleo, G., & Vicentini, F. (2023). Unbiasing time-dependent Variational Monte Carlo by projected quantum evolution. *Quantum*, *7*, 1131.

Time evolution: p-tVMC

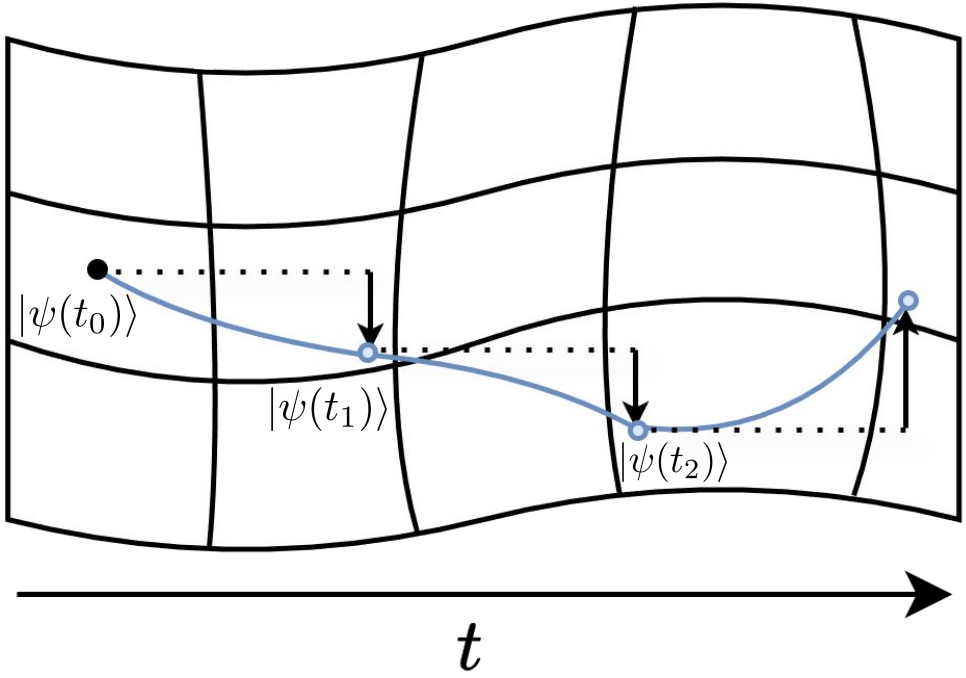
Minimize **Infidelity:**

$$I = 1 - \frac{|\langle\psi(t_1)|\psi(t_0)\rangle|^2}{\langle\psi(t_1)|\psi(t_1)\rangle\langle\psi(t_0)|\psi(t_1)\rangle}$$

$$|\psi(t)\rangle = e^{-iHt}|\psi(t_0)\rangle$$

$$|\psi(t_1)\rangle = (1 - iHdt)|\psi(t_0)\rangle$$

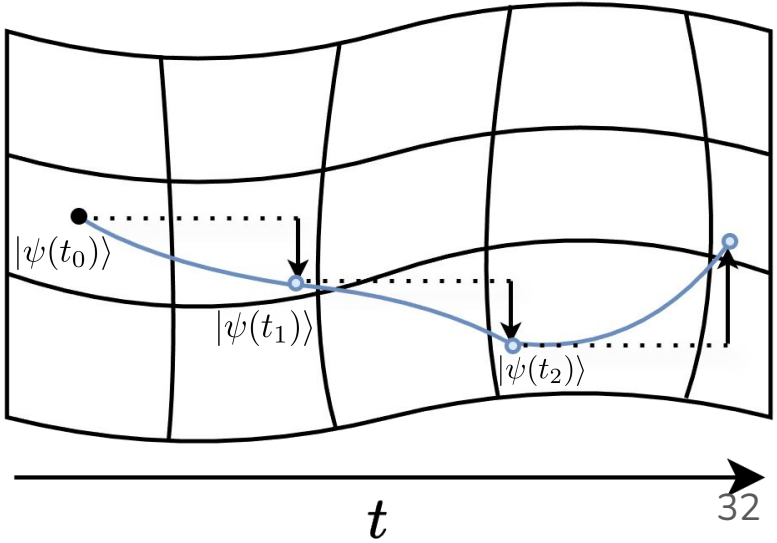
$$\equiv U$$



Sinibaldi, A., Giuliani, C., Carleo, G., & Vicentini, F. (2023). Unbiasing time-dependent Variational Monte Carlo by projected quantum evolution. *Quantum*, *7*, 1131.

Time evolution: p-tVMC

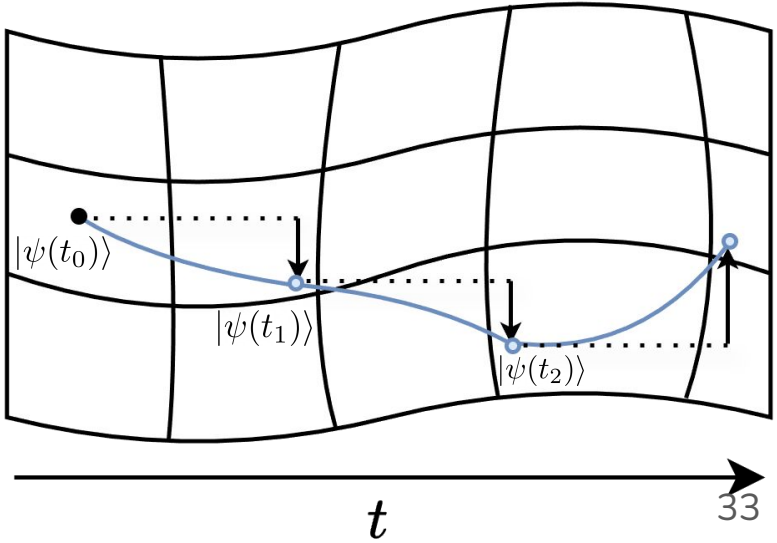
$$I = \mathbb{E}_{\sigma,\eta\sim p(\sigma|\Theta_1)p(\eta|\Theta_0)}\left[1 - \frac{\langle\sigma|U|\psi_{\Theta_0}(t_0)\rangle}{\langle\sigma|\psi_{\Theta_1}(t_1)\rangle}\frac{\langle\eta|U^\dagger|\psi_{\Theta_1}(t_1)\rangle}{\langle\eta|\psi_{\Theta_0}(t_0)\rangle}\right]$$



Time evolution: p-tVMC

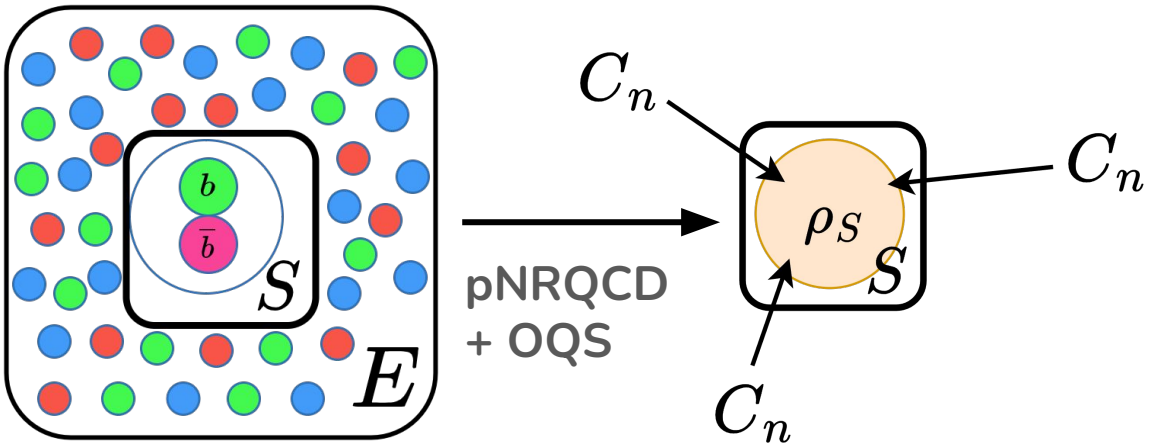
$$I = \mathbb{E}_{\sigma,\eta\sim p(\sigma|\Theta_1)p(\eta|\Theta_0)}\left[1 - \overbrace{\frac{\langle\sigma|U|\psi_{\Theta_0}(t_0)\rangle}{\langle\sigma|\psi_{\Theta_1}(t_1)\rangle}\frac{\langle\eta|U^\dagger|\psi_{\Theta_1}(t_1)\rangle}{\langle\eta|\psi_{\Theta_0}(t_0)\rangle}}^{\mathcal{I}_{\mathrm{loc}}}\right]$$

$$p(\sigma|\Theta_1)p(\eta|\Theta_0) = \frac{|\langle\sigma|\psi_{\Theta_1}\rangle|^2|\langle\eta|\psi_{\Theta_0}\rangle|^2}{\langle\psi_{\Theta_1}|\psi_{\Theta_1}\rangle\langle\psi_{\Theta_0}|\psi_{\Theta_0}\rangle}$$



Actually, one can find a Control Variate for this specific estimator, which is important to obtain a good SNR for $\mathcal{I}_{\mathrm{loc}} \to 0$

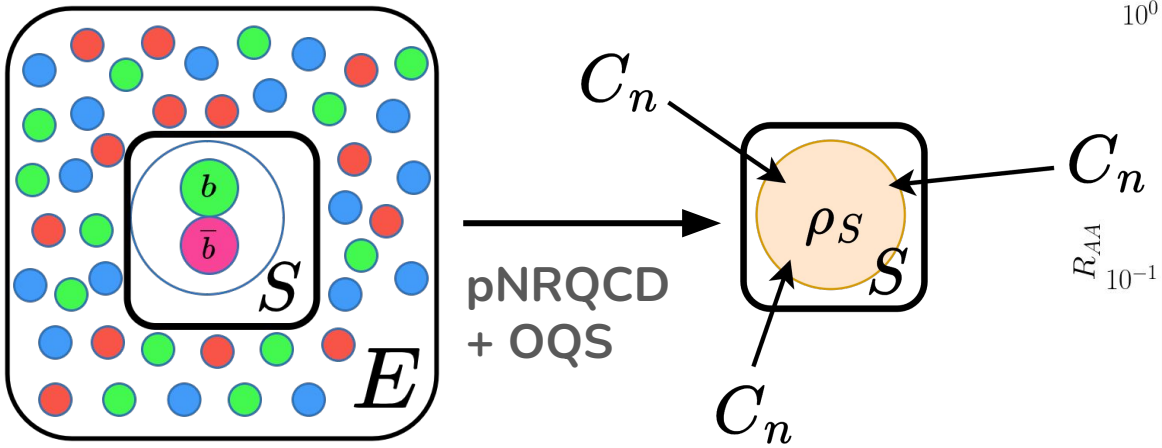
Quarkonium Suppression Simulator

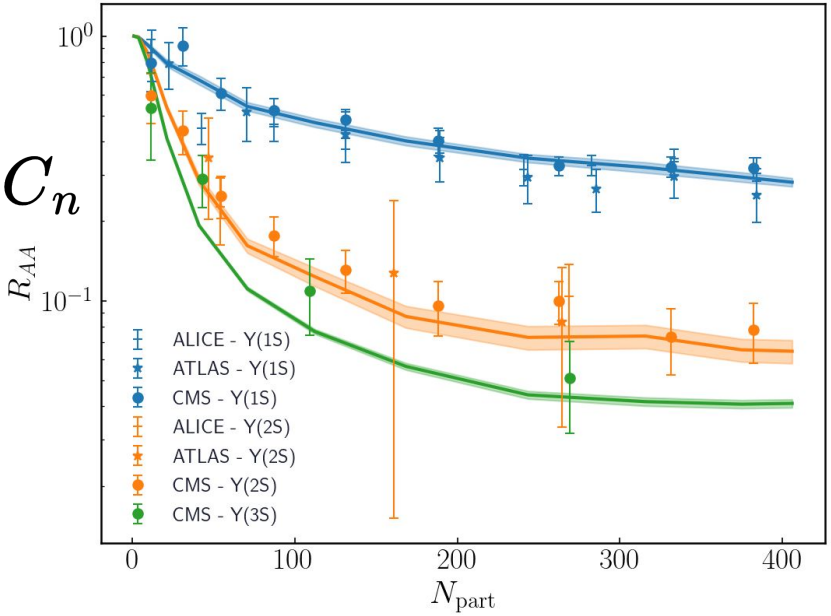


$$\frac{d\rho_S}{dt} = -i[H_S, \rho_S] + \sum_n \left(C_n \rho_S C_n^\dagger - \frac{1}{2} \left\{ C_n^\dagger C_n, \rho_S \right\} \right)$$

Very large matrix

Quarkonium Suppression Simulator





$$\frac{d\rho_S}{dt} = -i[H_S, \rho_S] + \sum_n \left(C_n \rho_S C_n^\dagger - \frac{1}{2} \left\{ C_n^\dagger C_n, \rho_S \right\} \right)$$

Very large matrix

Learning the Lindblad equation

Schrödinger equation

$$\frac{d}{dt}|\psi\rangle = -iH|\psi\rangle$$

$$|\psi(t)\rangle = e^{-iHt}|\psi(0)\rangle$$

Lindblad equation

$$\frac{d}{dt}\rho = \mathcal{L}\rho$$

$$\rho(t) = e^{\mathcal{L}t}\rho(0)$$

Learning the Lindblad equation

Since $e^{\mathcal{L}t}$ is non unitary the actual time evolution requires to sample from a different distribution which we do by reweighting

Schrödinger equation

$$\frac{d}{dt}|\psi\rangle = -iH|\psi\rangle$$

$$|\psi(t)\rangle = e^{-iHt}|\psi(0)\rangle$$

$$\mathcal{I}_{\text{loc}} = 1 - \frac{\langle\sigma|\psi_0\rangle}{\langle\sigma|\psi_1\rangle}\frac{\langle\eta|\psi_1\rangle}{\langle\eta|\psi_0\rangle}$$

$$p(\sigma|\Theta_1)p(\eta|\Theta_0) = \frac{|\langle\sigma|\psi_{\Theta_1}\rangle|^2|\langle\eta|\psi_{\Theta_0}\rangle|^2}{\langle\psi_{\Theta_1}|\psi_{\Theta_1}\rangle\langle\psi_{\Theta_0}|\psi_{\Theta_0}\rangle}$$

Lindblad equation

$$\frac{d}{dt}\rho = \mathcal{L}\rho$$

$$\rho(t) = e^{\mathcal{L}t}\rho(0)$$

$$\mathcal{I}_{\text{loc}} = 1 - \frac{\langle\sigma|\rho_0|\eta\rangle}{\langle\sigma|\rho_1|\eta\rangle}\frac{\langle\omega|\rho_1|\zeta\rangle}{\langle\omega|\rho_0|\zeta\rangle}$$

$$p(\sigma,\eta)p(\omega,\zeta) = \frac{|\langle\sigma|\rho_2|\eta\rangle|^2}{\sum_{\eta\sigma}|\langle\sigma|\rho_2|\eta\rangle|^2}\frac{|\langle\omega|\rho_1|\zeta\rangle|^2}{\sum_{\omega\zeta}|\langle\omega|\rho_1|\zeta\rangle|^2}$$

Density matrices

1. $\mathrm{Tr}\rho = 1$
2. $\rho^\dagger = \rho$
3. $\rho \geq 0$

Enforce in a neural network!

Density matrices

1. $\mathrm{Tr}\rho = 1$
2. $\rho^\dagger = \rho$
3. $\rho \geq 0$

Enforce in a neural network!

$$\langle\sigma|\,\hat{\rho}\,|\eta\rangle = \prod_{h=1}^{K}\sum_{a=1}^{R}\psi^{(h)}_{\sigma,a}\,[\psi^{(h)}_{\eta,a}]^*$$

Vicentini, F., Rossi, R., & Carleo, G. (2022). Positive-definite parametrization of mixed quantum states with deep neural networks. *arXiv:2206.13488*.

Density matrices

To evaluate expectation values we have to sample from

$$p(\sigma, \eta) = \frac{1}{N} |\langle \sigma | \rho | \eta \rangle|^2$$

Vicentini, F., Rossi, R., & Carleo, G. (2022). Positive-definite parametrization of mixed quantum states with deep neural networks. *arXiv:2206.13488*.

Density matrices

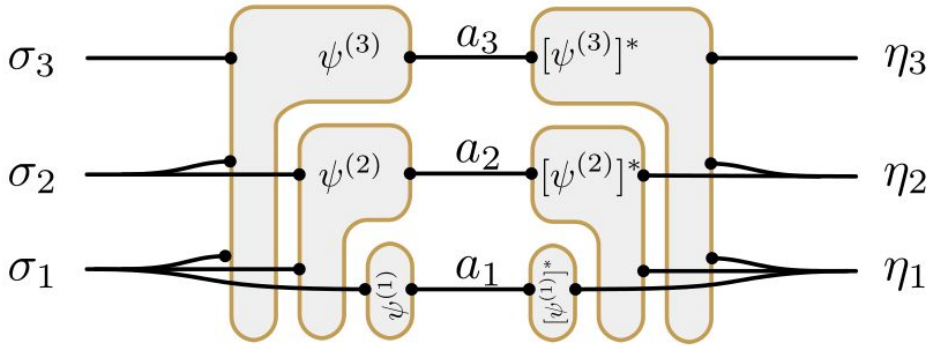
Since we are simulating a particle in position space we implement binary encodings of position basis states

To evaluate expectation values we have to sample from

$$p(\sigma, \eta) = \frac{1}{N} |\langle \sigma | \rho | \eta \rangle|^2$$

Restrict to autoregressive structure:

$$\langle \sigma | \hat{\rho} | \eta \rangle = \prod_{h=1}^{N} \sum_{a=1}^{R} \psi_{\sigma_{\leq h}, a} \left[\psi_{\eta_{\leq h}, a} \right]^*$$



Vicentini, F., Rossi, R., & Carleo, G. (2022). Positive-definite parametrization of mixed quantum states with deep neural networks. *arXiv:2206.13488*.

Simulation:

$$H = \frac{p^2}{M} - \frac{\alpha}{x} + \beta\{x, p\} \quad C = x + \gamma i p$$

1. **Initialization:** Learn some ρ_0

2. **Time evolution:**

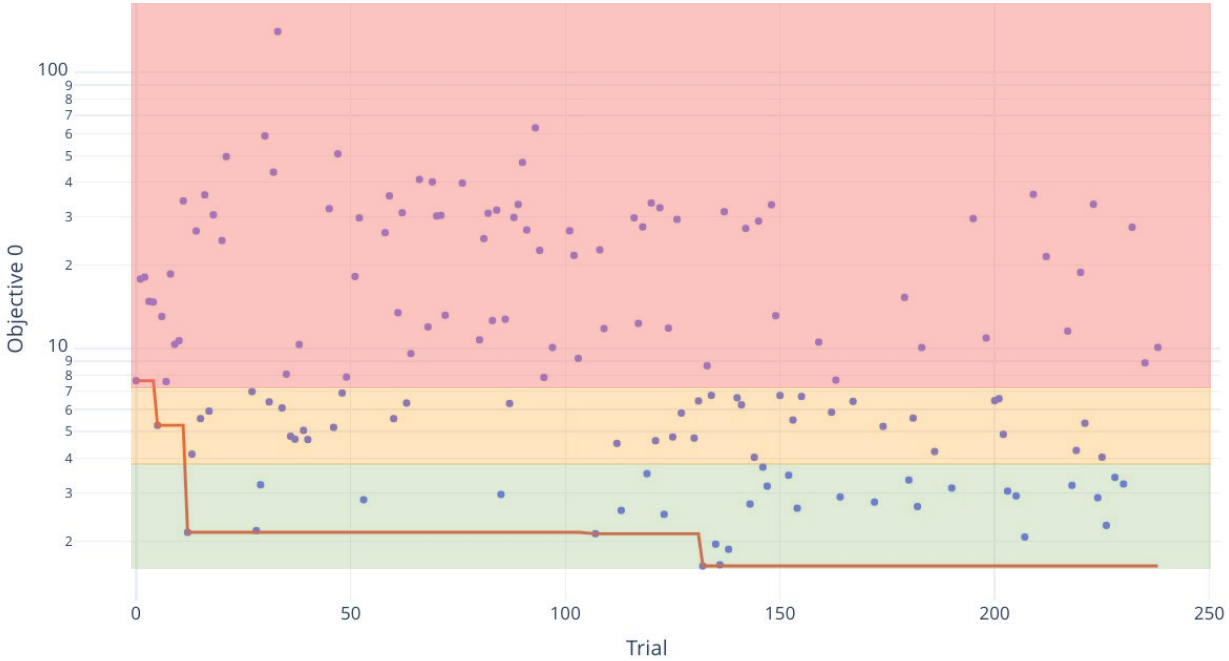
   a. Sample states $\sigma, \eta, \omega, \zeta$ from ρ and $(1 + \delta t \mathcal{L})\rho$

   b. Evaluate $\mathcal{I}_{\text{loc}}(\sigma, \eta, \omega, \zeta) = 1 - \frac{\langle\sigma|(1+\delta t\mathcal{L})\rho_1|\eta\rangle \langle\omega|\rho_2|\zeta\rangle}{\langle\sigma|\rho_2|\eta\rangle \langle\omega|(1+\delta t\mathcal{L})\rho_1|\zeta\rangle}$ and the gradient

   c. Copy model $\rho_1 \leftarrow \rho_2$

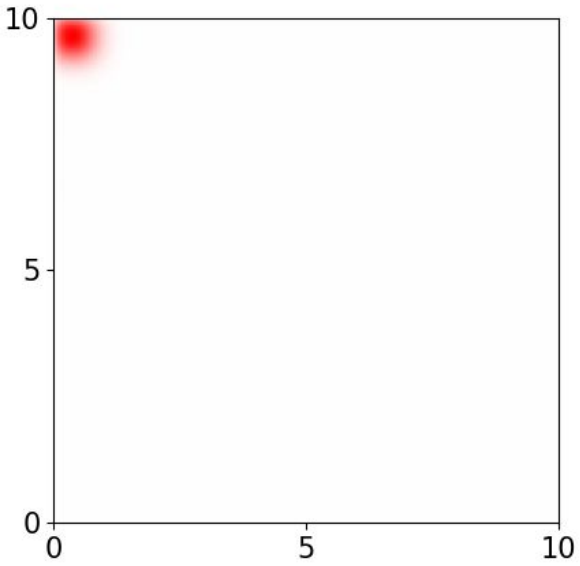
To get better convergence in δt , we designed a special estimator which implements an implicit second order Crank-Nicolson solver, for $e^{\mathcal{L}t}$ including a suitable Control Variate.

Hyperparameters

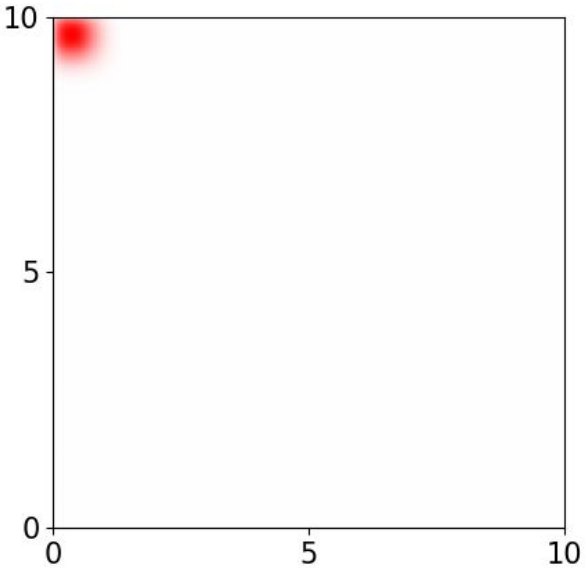


Simulation results

Monte Carlo Wavefunction

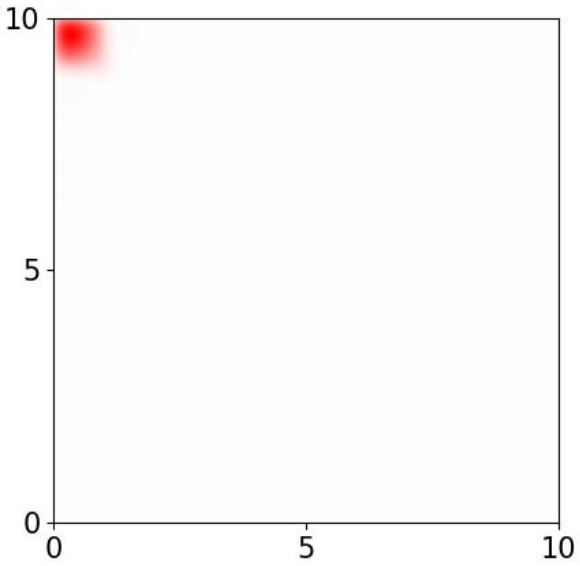


Neural Network

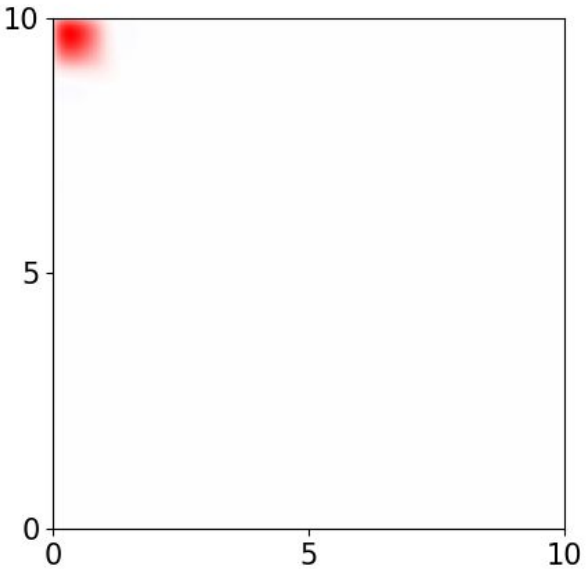


Simulation results

Monte Carlo Wavefunction

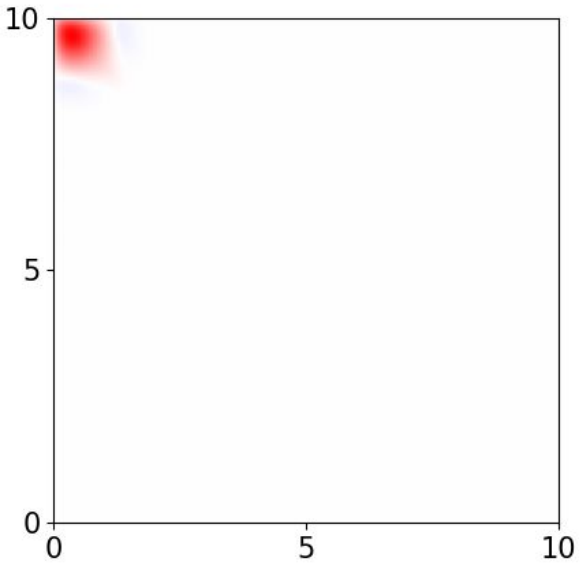


Neural Network

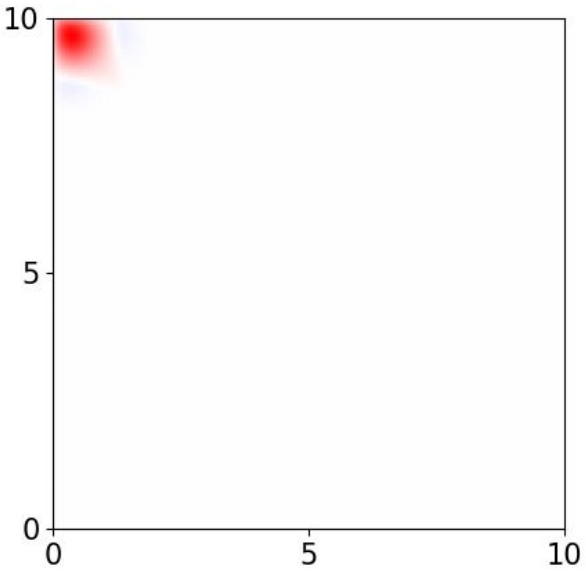


Simulation results

Monte Carlo Wavefunction

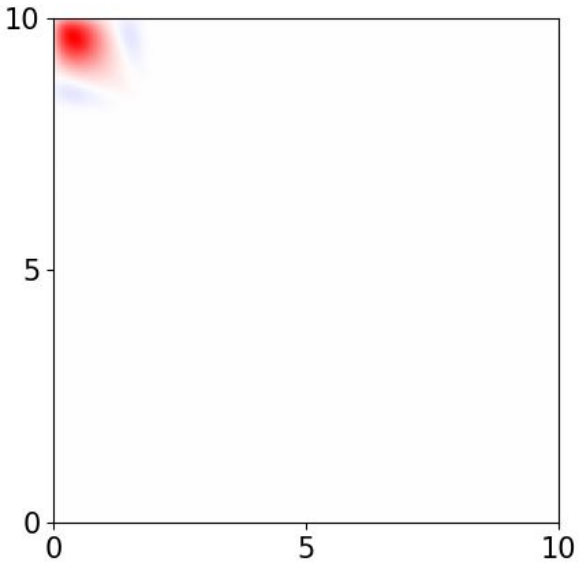


Neural Network

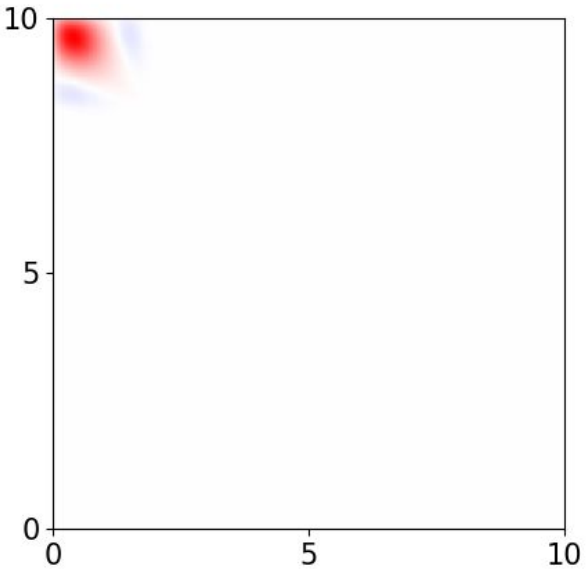


Simulation results

Monte Carlo Wavefunction

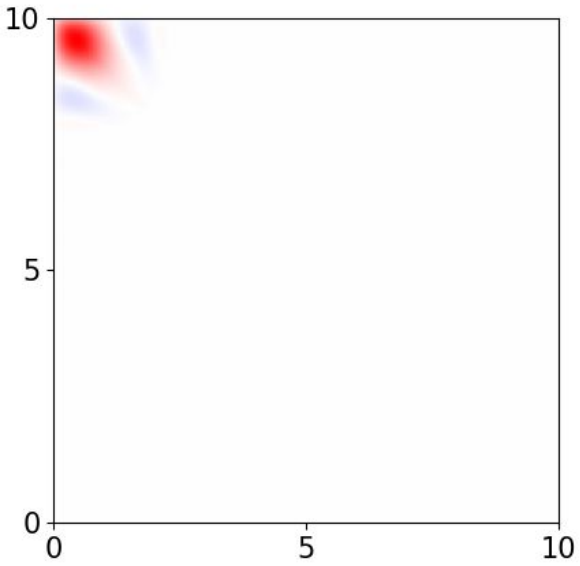


Neural Network

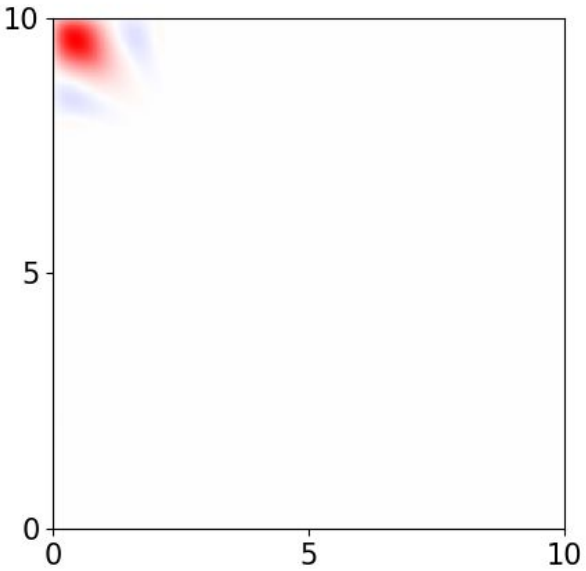


Simulation results

Monte Carlo Wavefunction

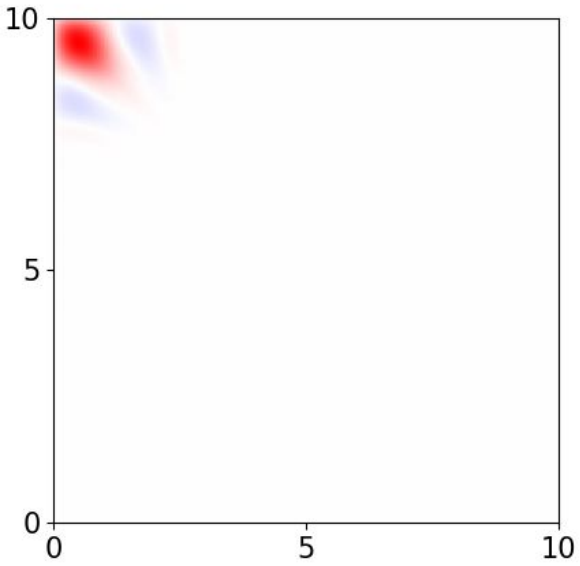


Neural Network

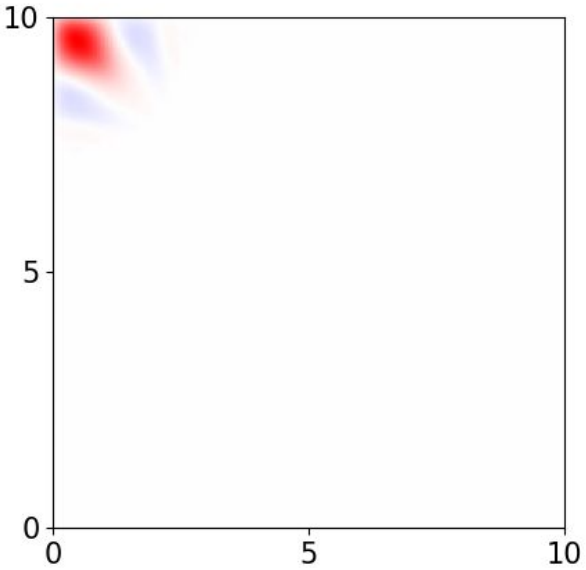


Simulation results

Monte Carlo Wavefunction

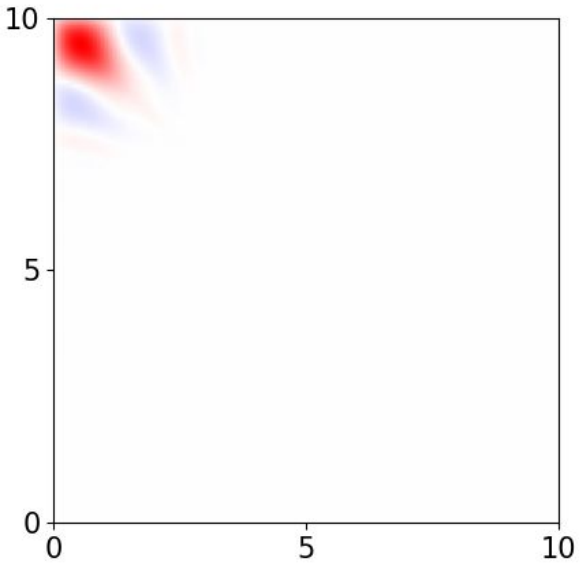


Neural Network

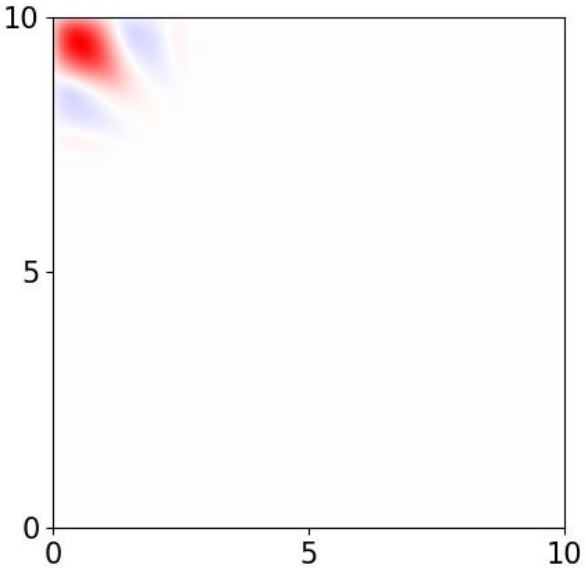


Simulation results

Monte Carlo Wavefunction

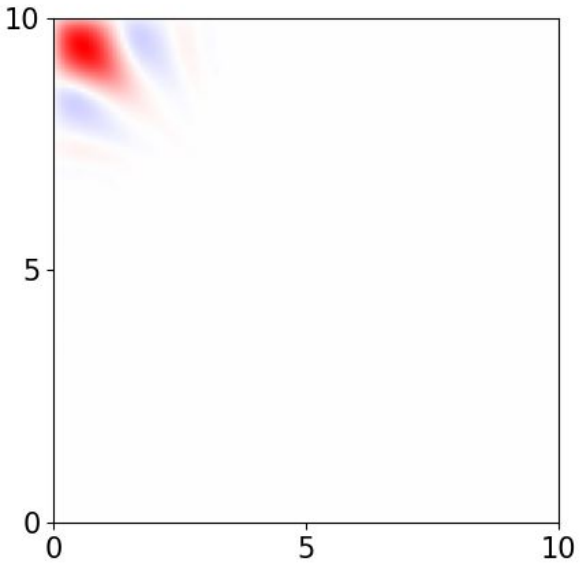


Neural Network

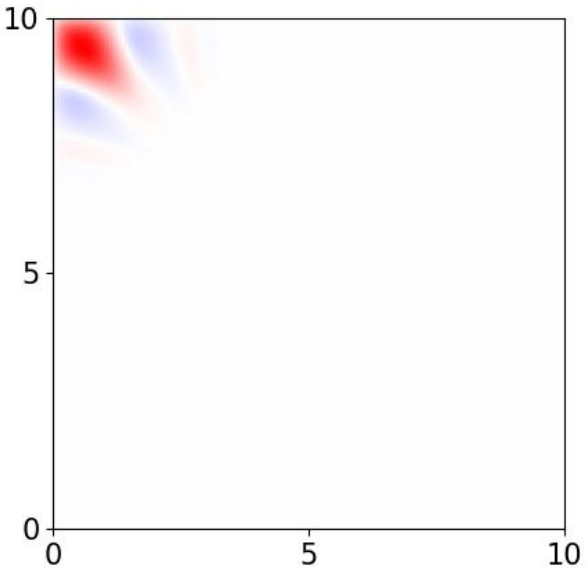


Simulation results

Monte Carlo Wavefunction

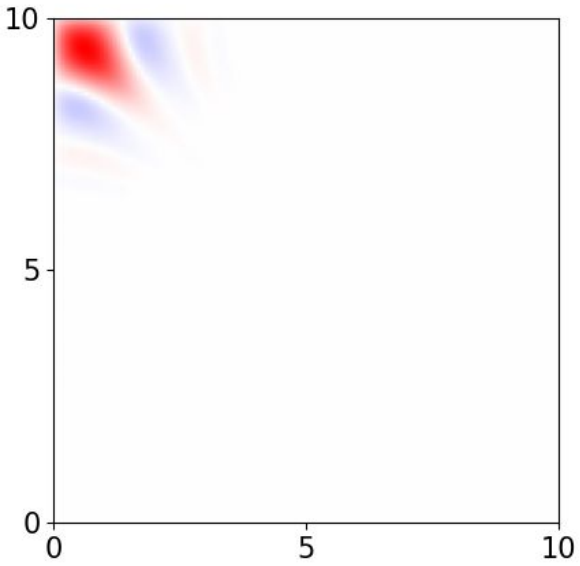


Neural Network

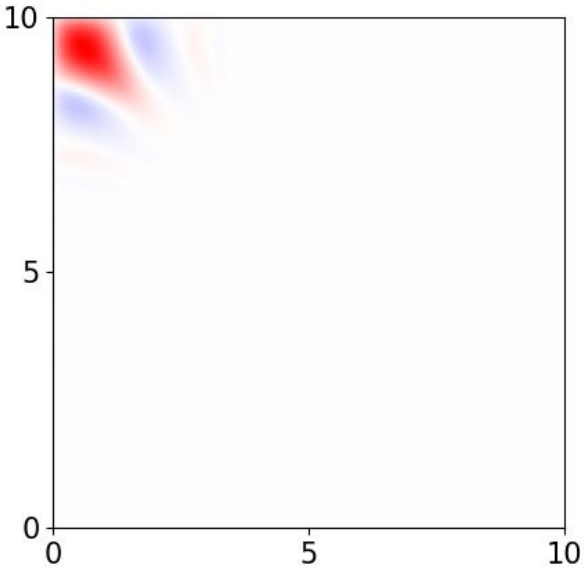


Simulation results

Monte Carlo Wavefunction

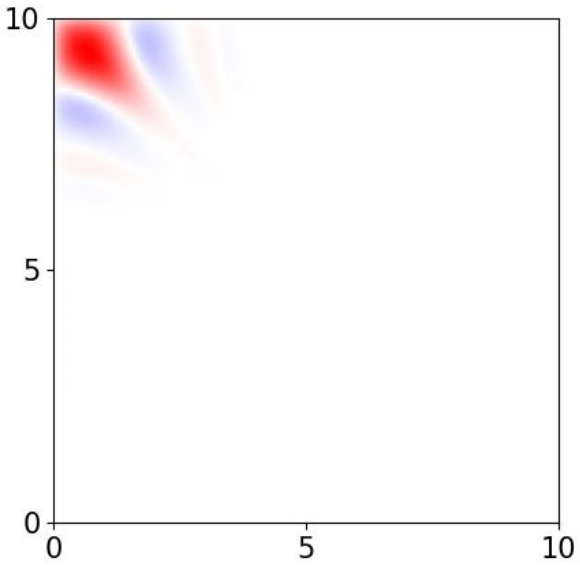


Neural Network

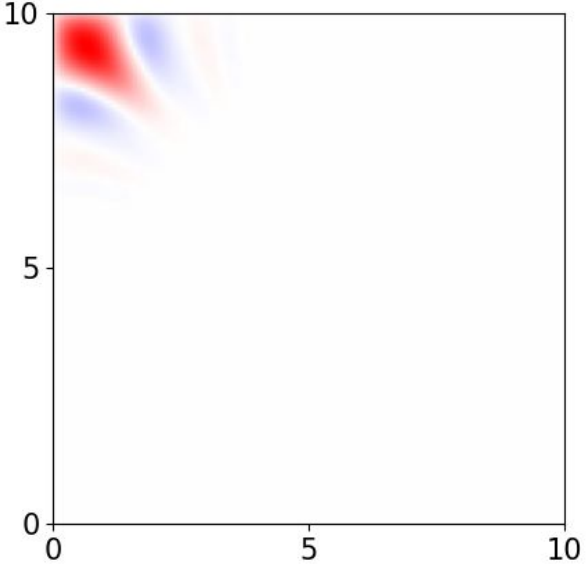


Simulation results

Monte Carlo Wavefunction

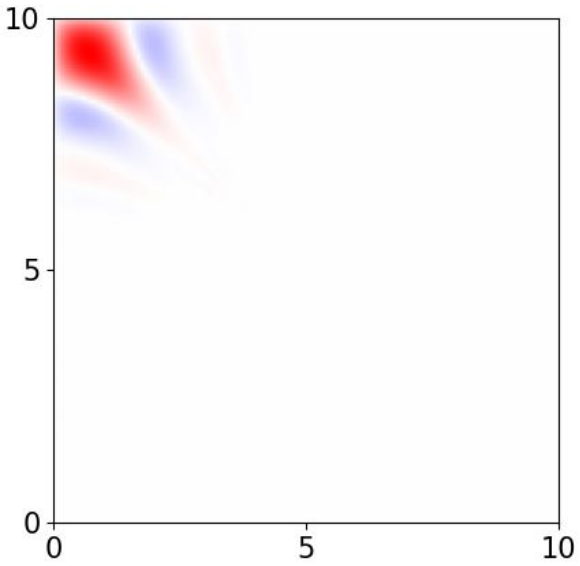


Neural Network

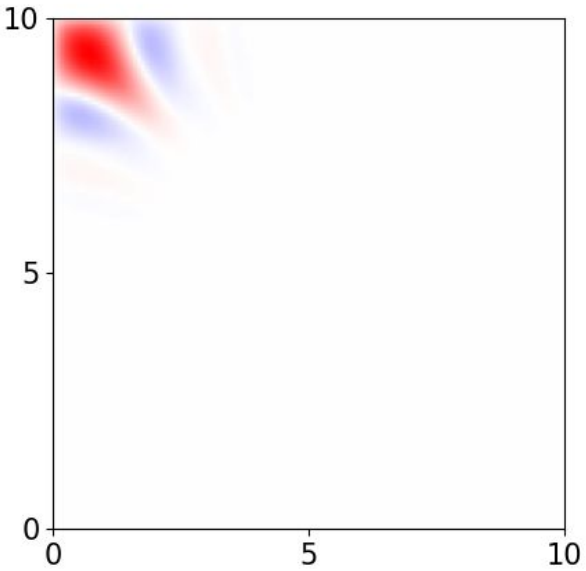


Simulation results

Monte Carlo Wavefunction

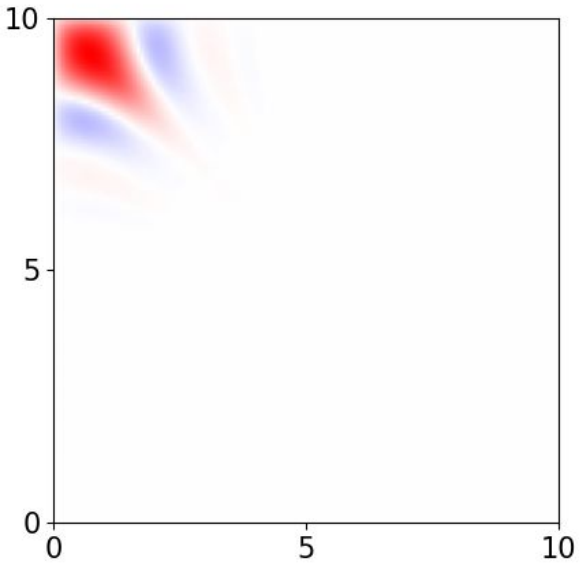


Neural Network

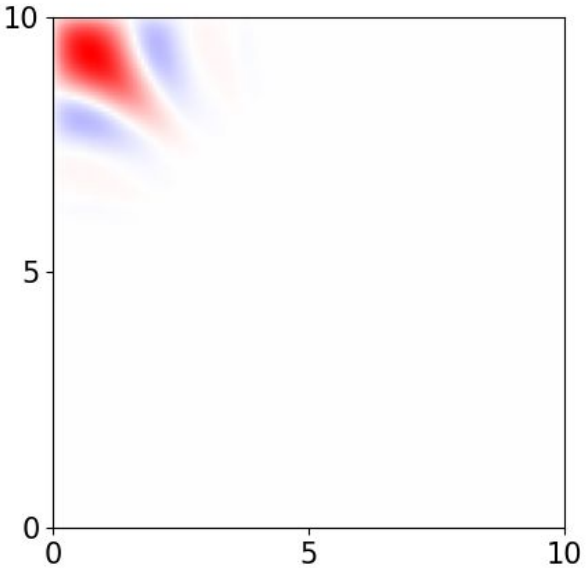


Simulation results

Monte Carlo Wavefunction

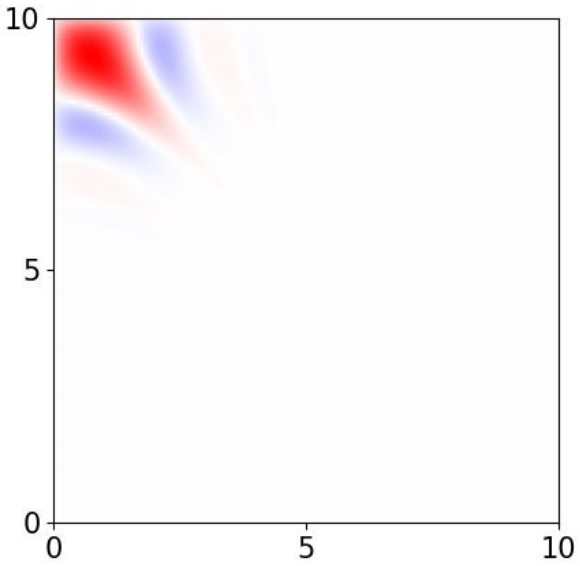


Neural Network

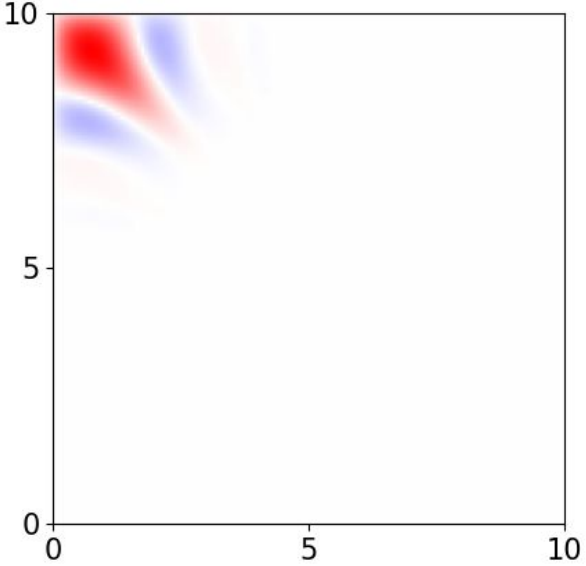


Simulation results

Monte Carlo Wavefunction

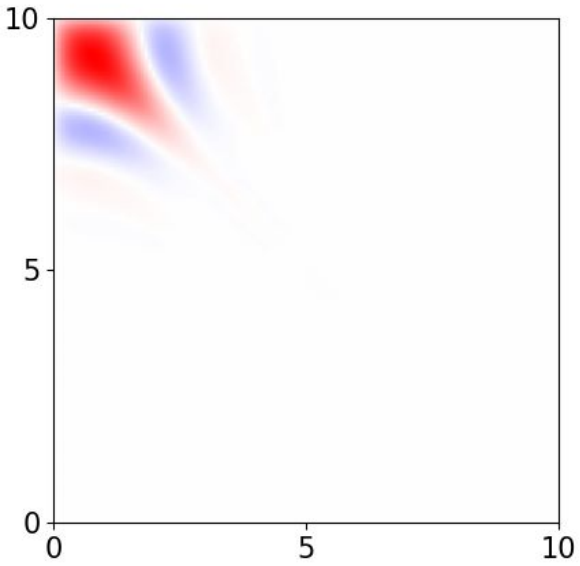


Neural Network

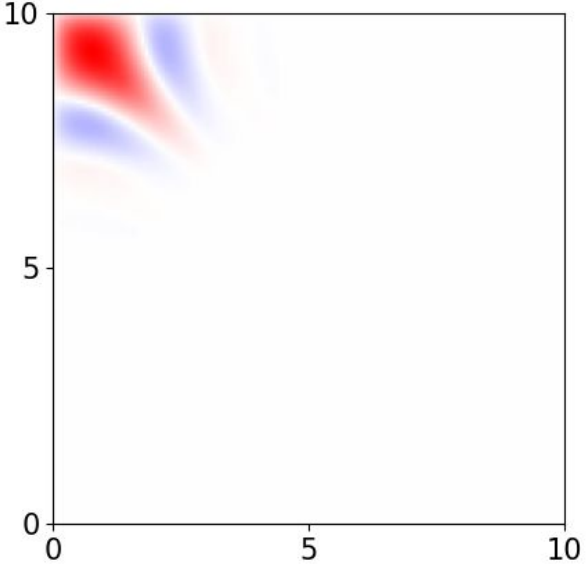


Simulation results

Monte Carlo Wavefunction

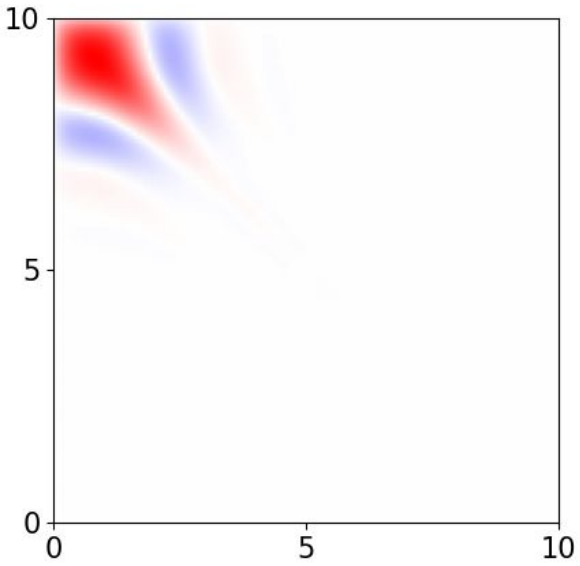


Neural Network

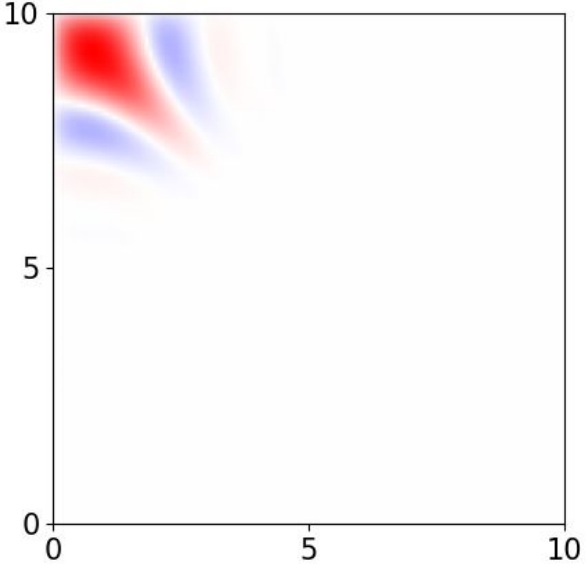


Simulation results

Monte Carlo Wavefunction

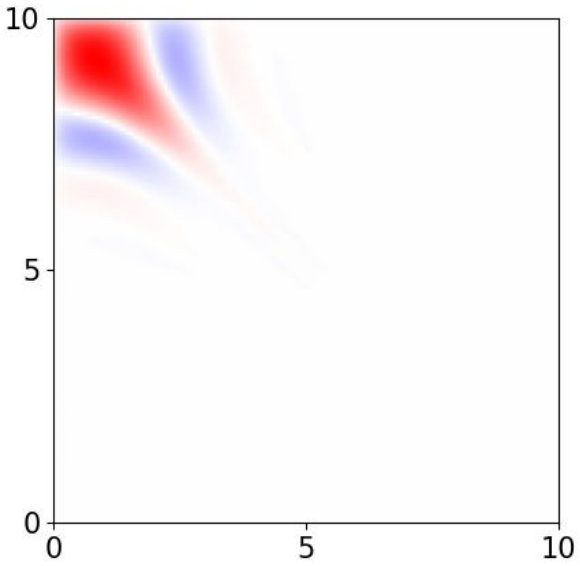


Neural Network

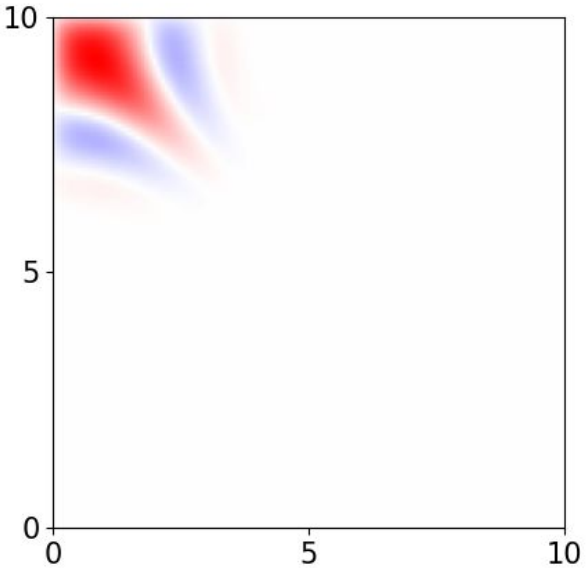


Simulation results

Monte Carlo Wavefunction

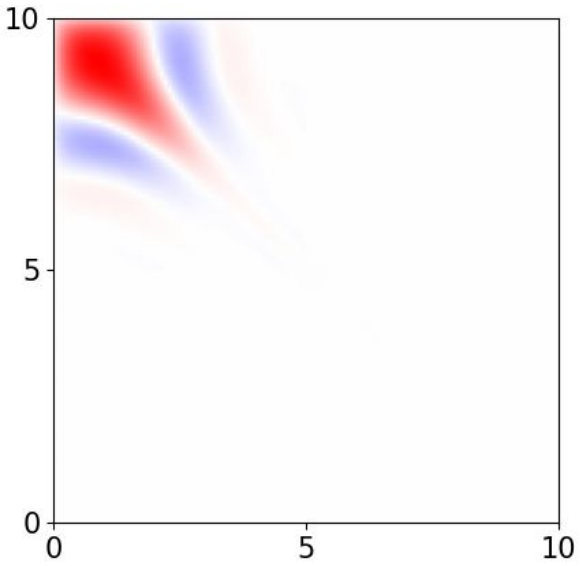


Neural Network

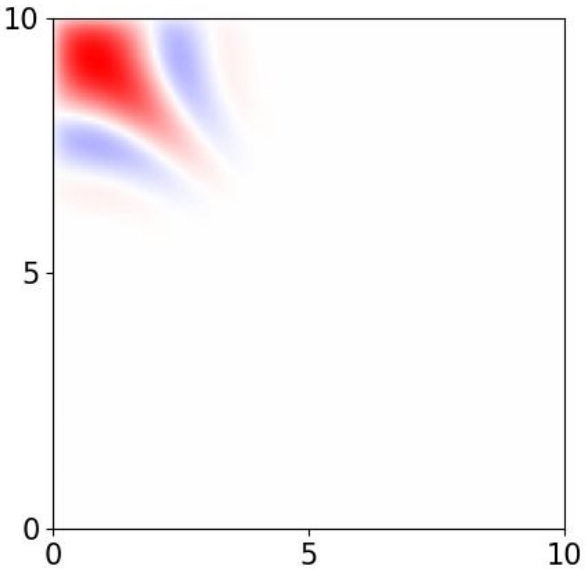


Simulation results

Monte Carlo Wavefunction

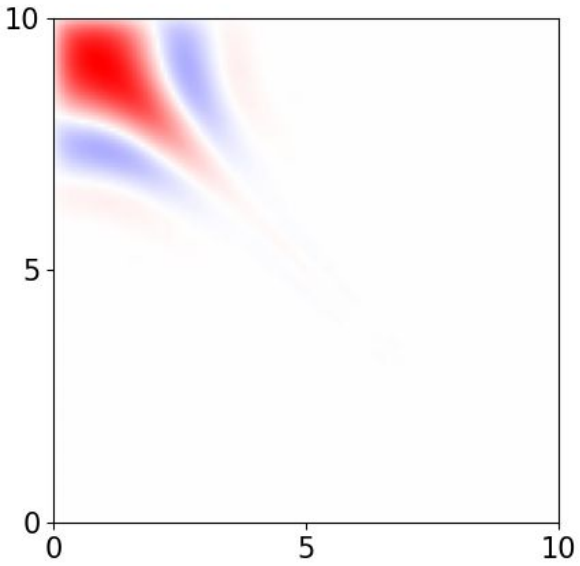


Neural Network

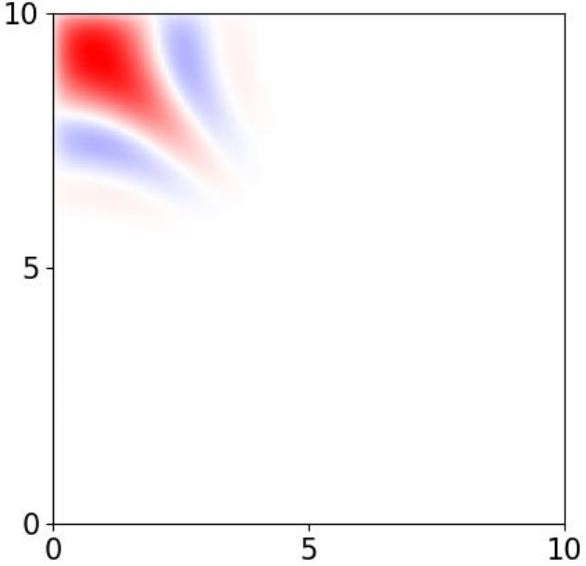


Simulation results

Monte Carlo Wavefunction

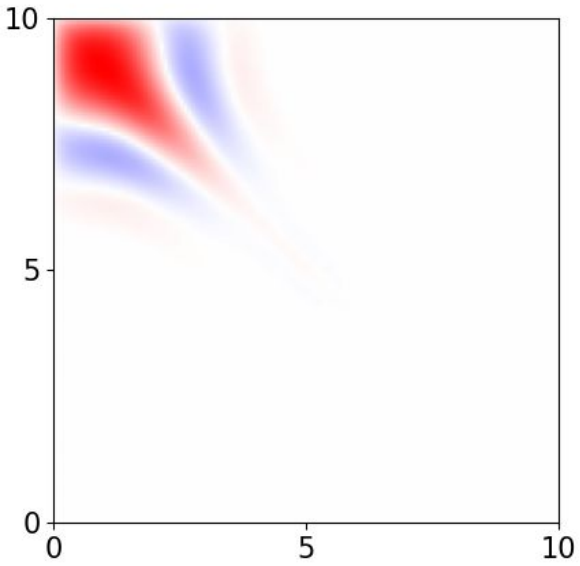


Neural Network

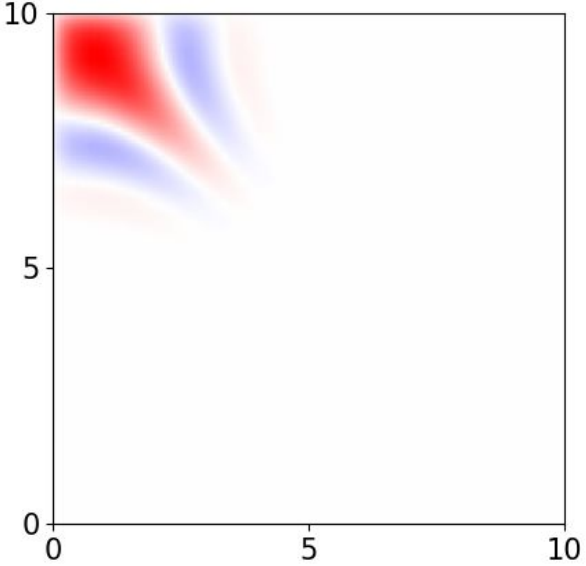


Simulation results

Monte Carlo Wavefunction

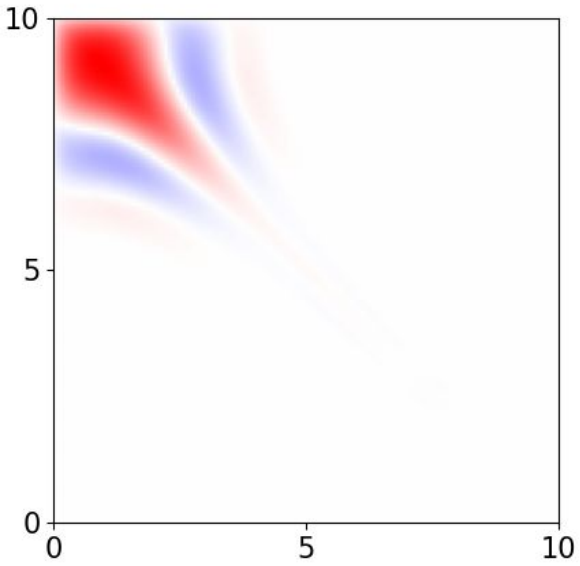


Neural Network

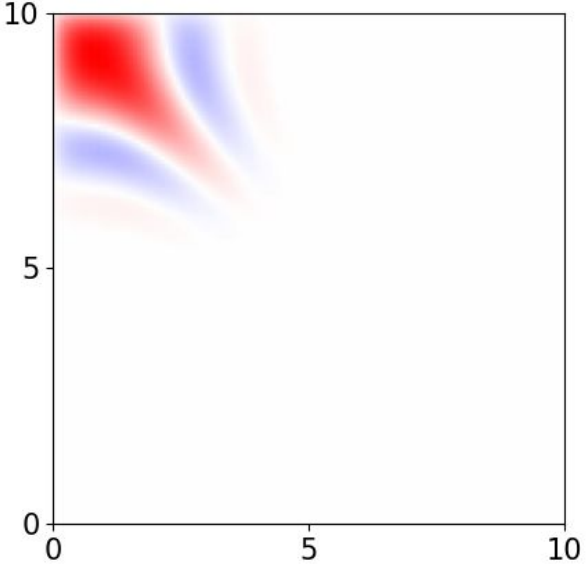


Simulation results

Monte Carlo Wavefunction

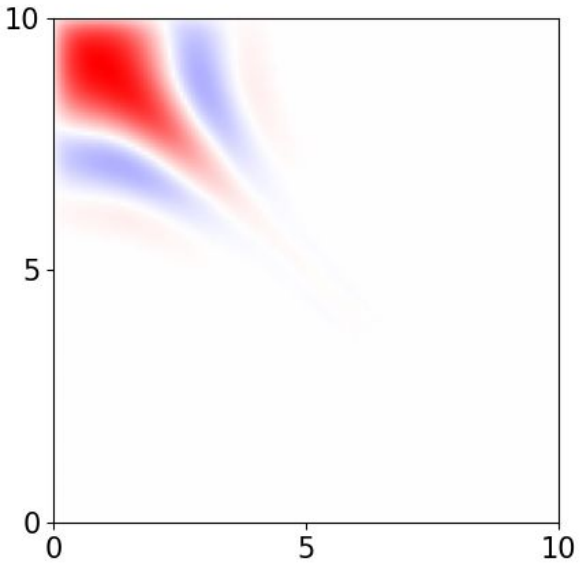


Neural Network

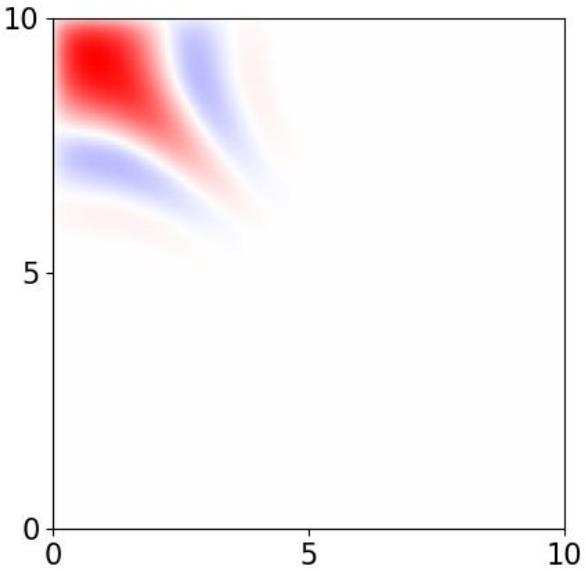


Simulation results

Monte Carlo Wavefunction

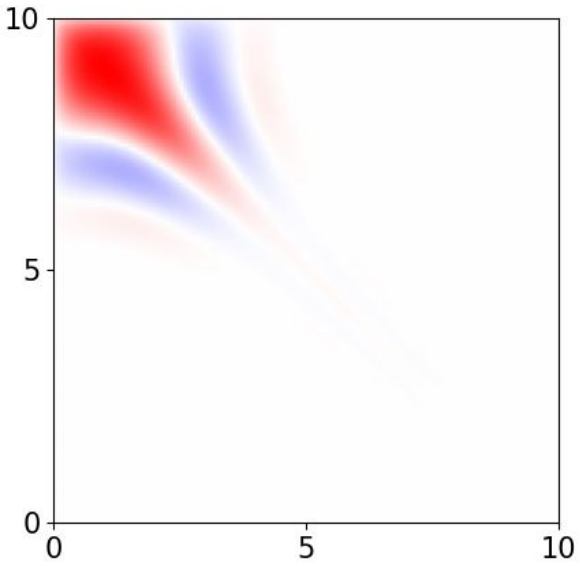


Neural Network

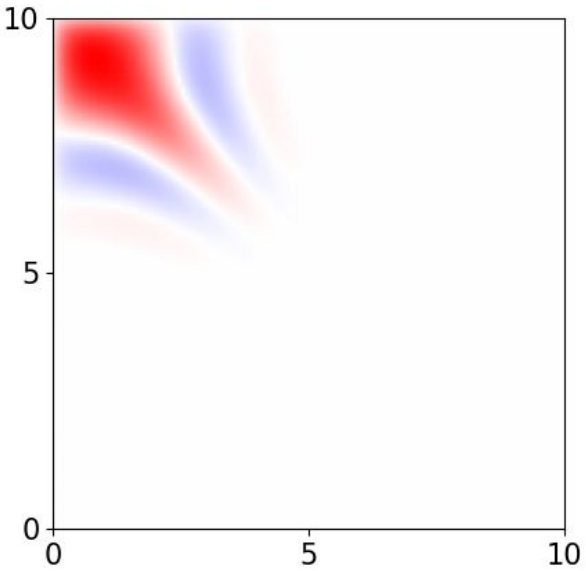


Simulation results

Monte Carlo Wavefunction

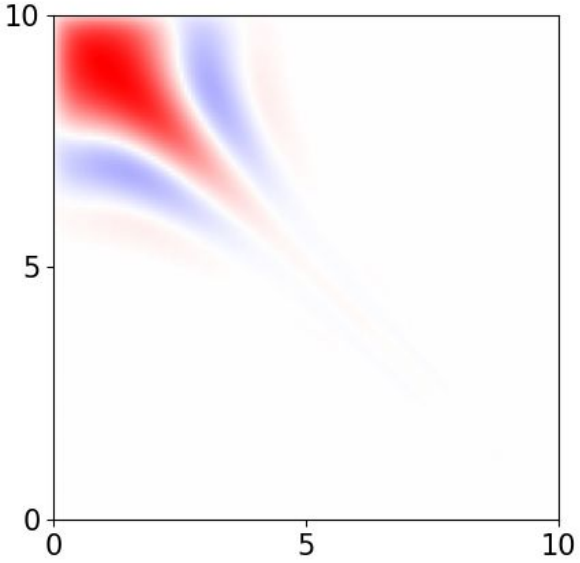


Neural Network

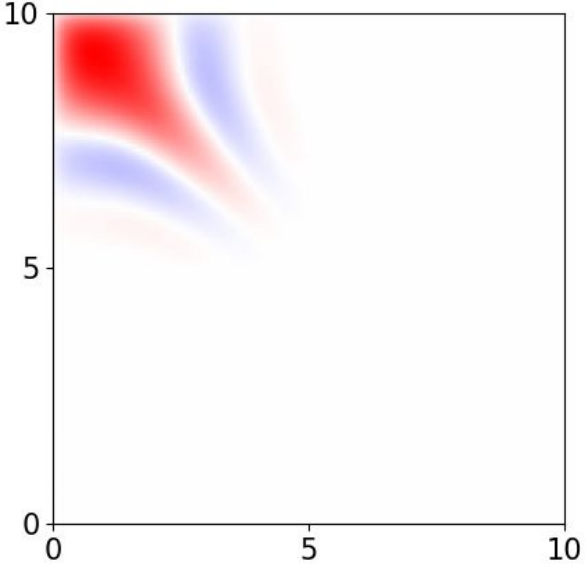


Simulation results

Monte Carlo Wavefunction

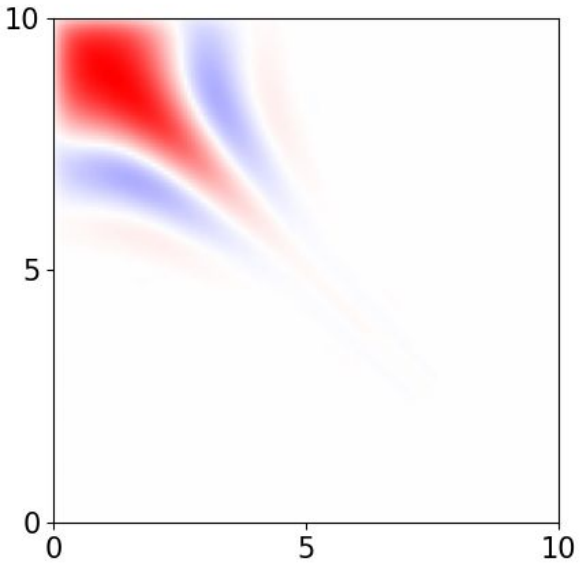


Neural Network

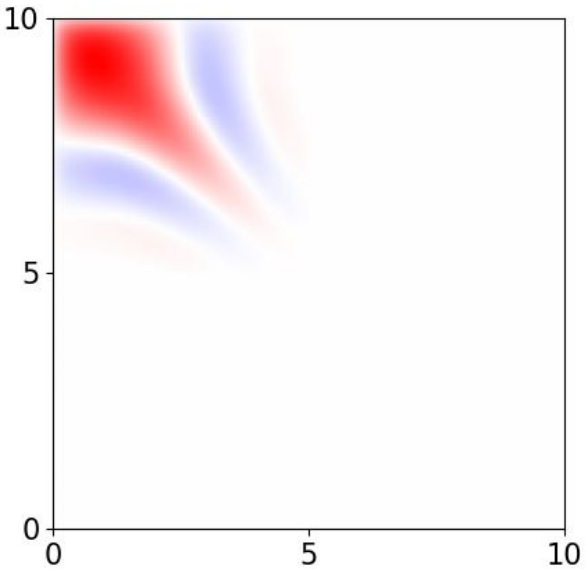


Simulation results

Monte Carlo Wavefunction

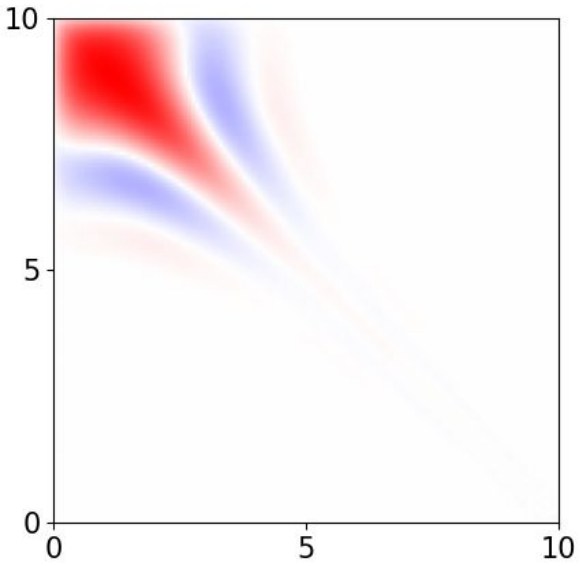


Neural Network

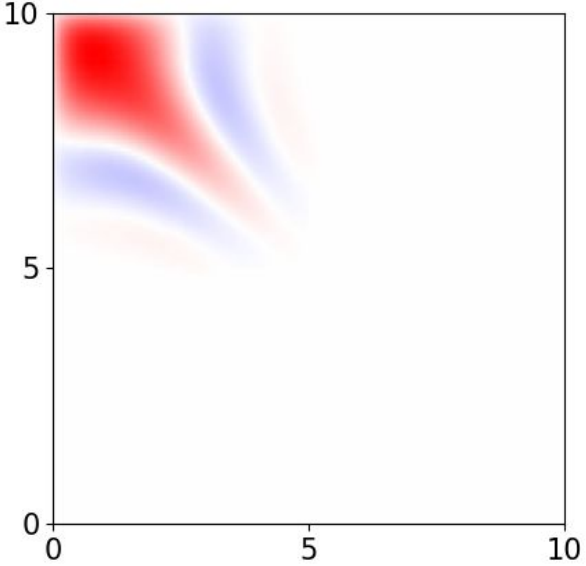


Simulation results

Monte Carlo Wavefunction

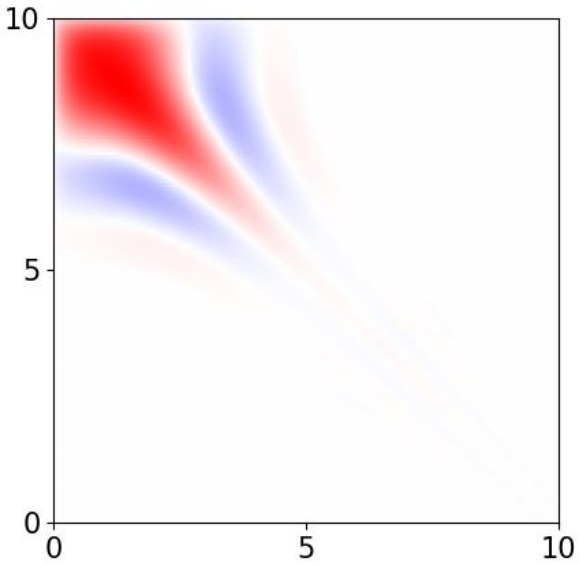


Neural Network

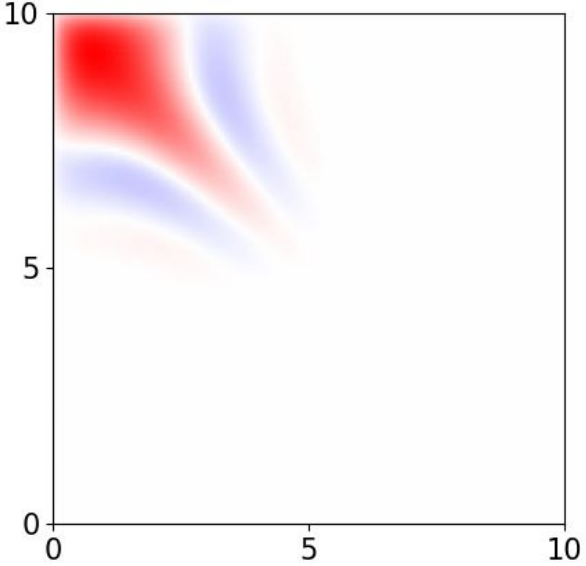


Simulation results

Monte Carlo Wavefunction

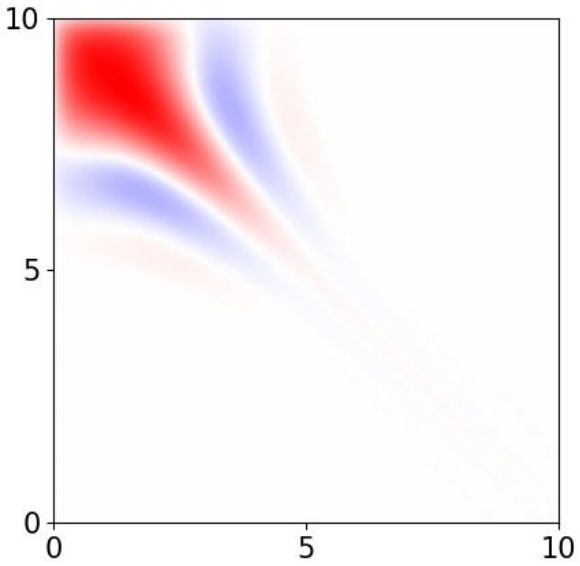


Neural Network

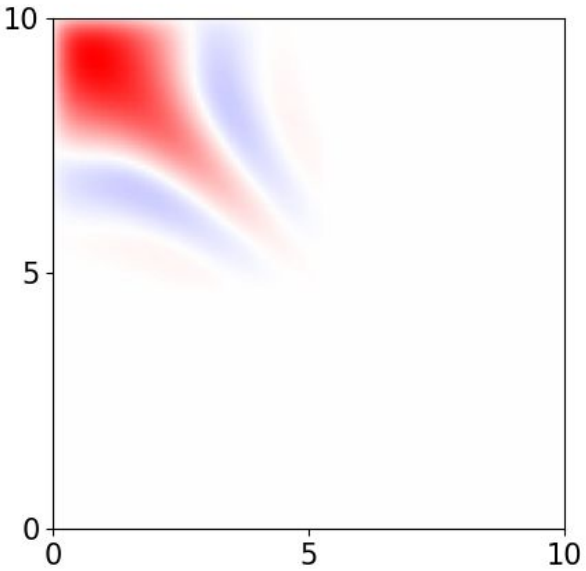


Simulation results

Monte Carlo Wavefunction

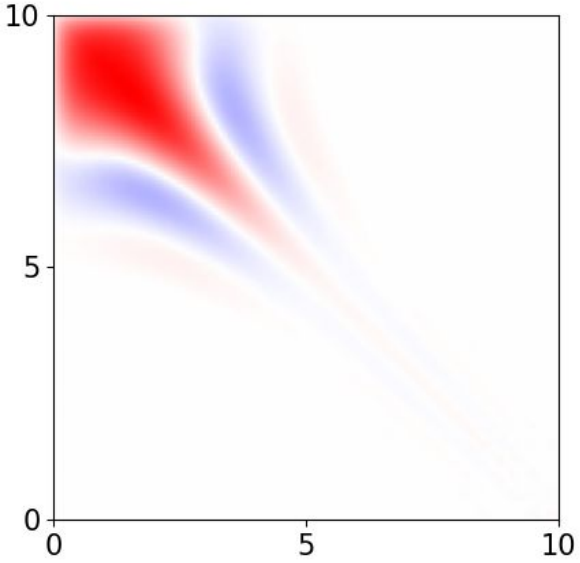


Neural Network

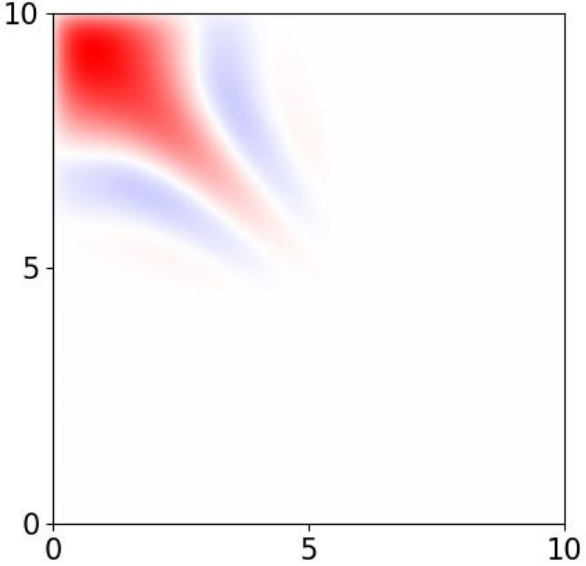


Simulation results

Monte Carlo Wavefunction

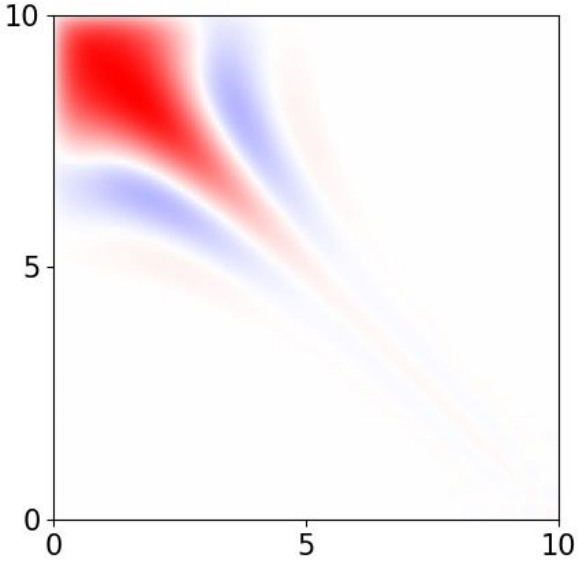


Neural Network

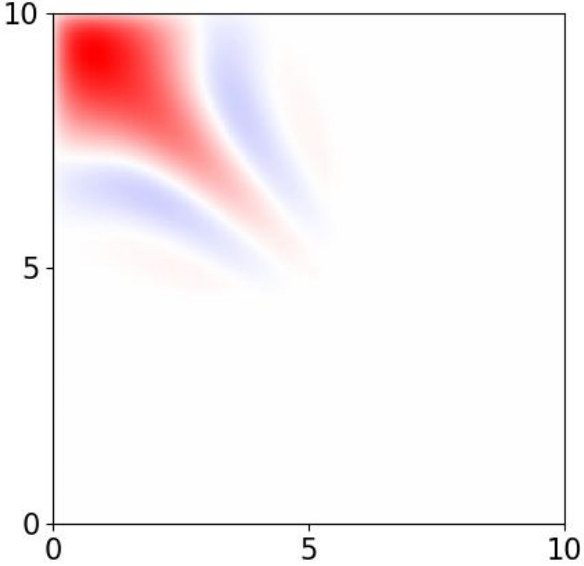


Simulation results

Monte Carlo Wavefunction

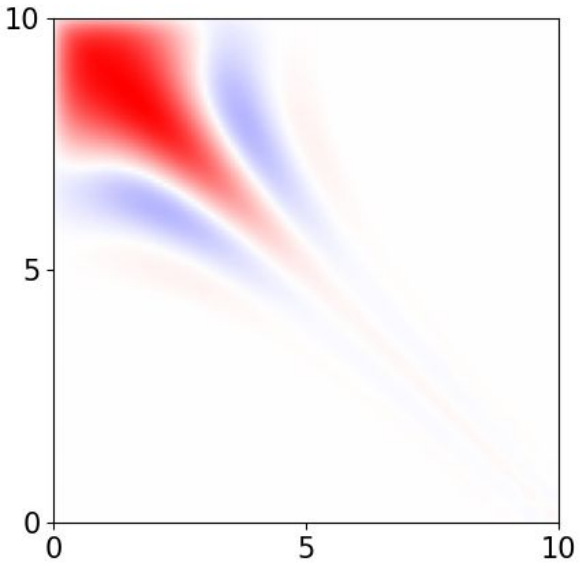


Neural Network

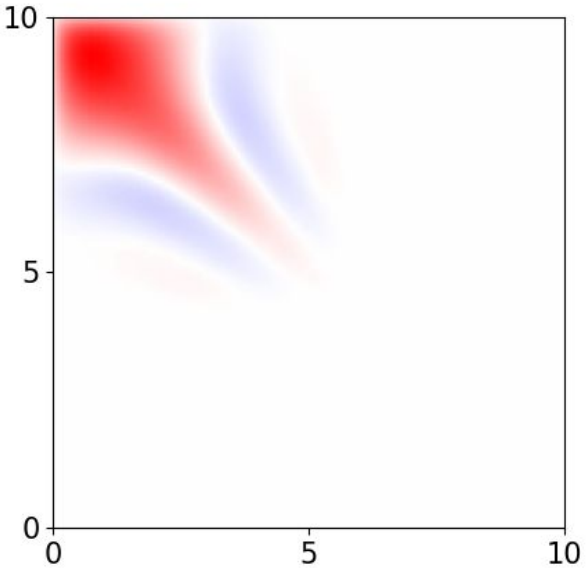


Simulation results

Monte Carlo Wavefunction

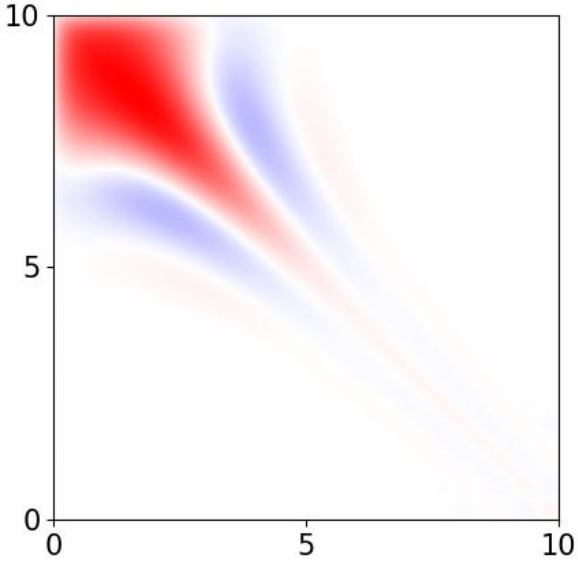


Neural Network

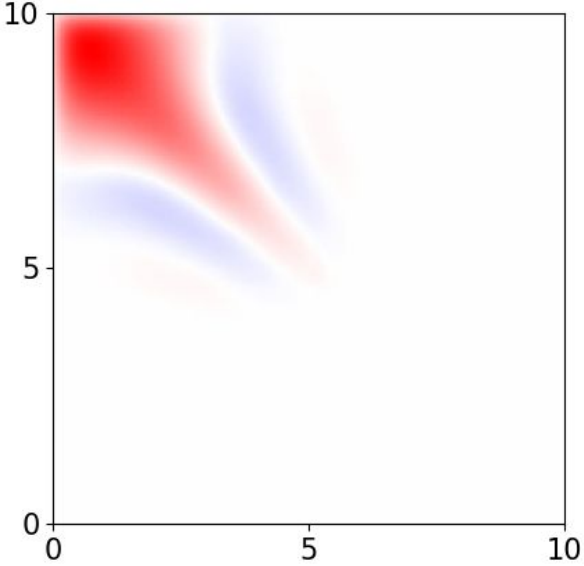


Simulation results

Monte Carlo Wavefunction

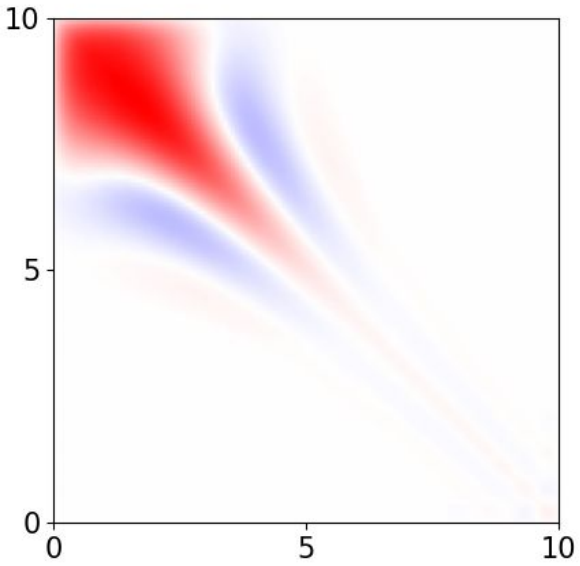


Neural Network

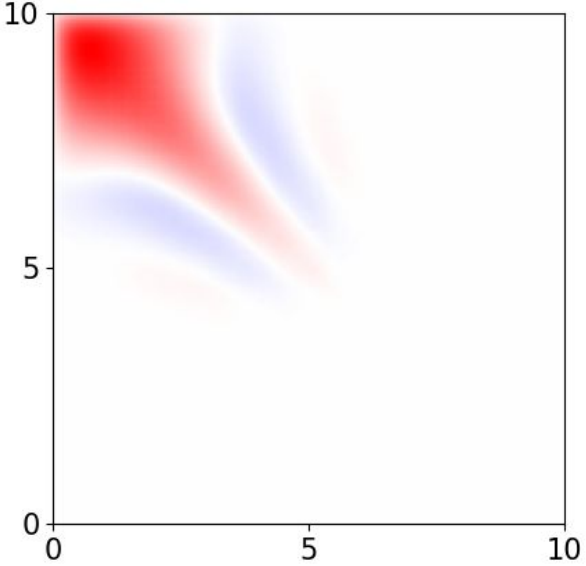


Simulation results

Monte Carlo Wavefunction

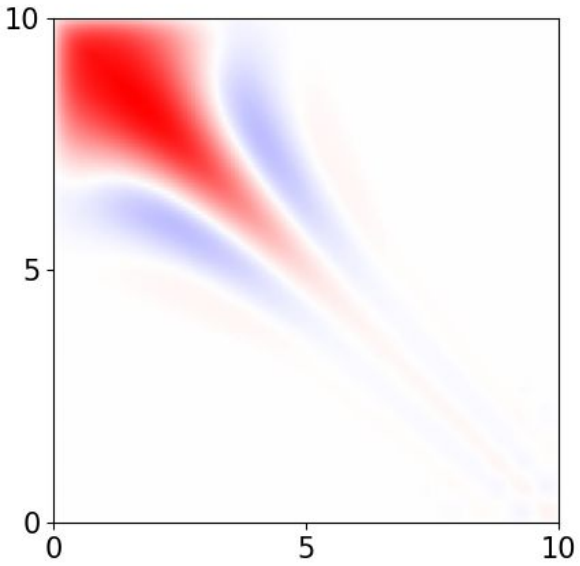


Neural Network

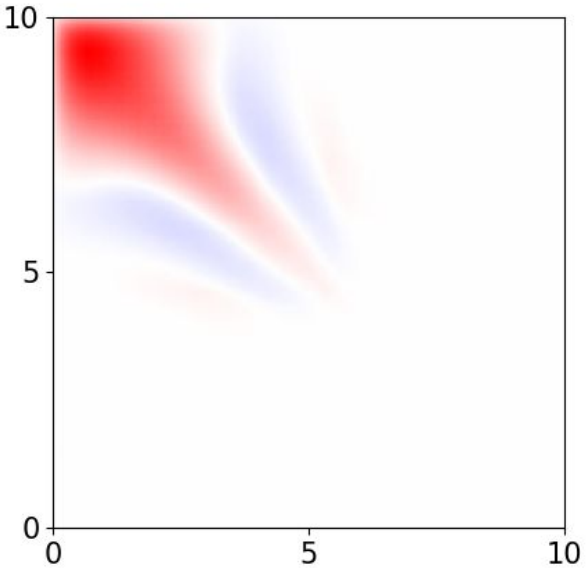


Simulation results

Monte Carlo Wavefunction

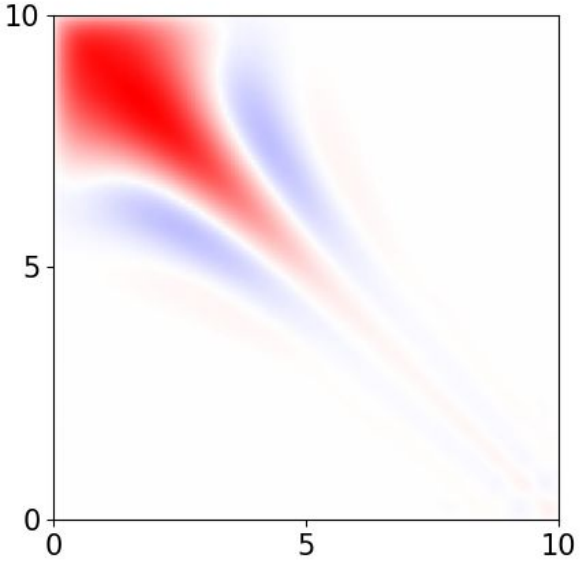


Neural Network

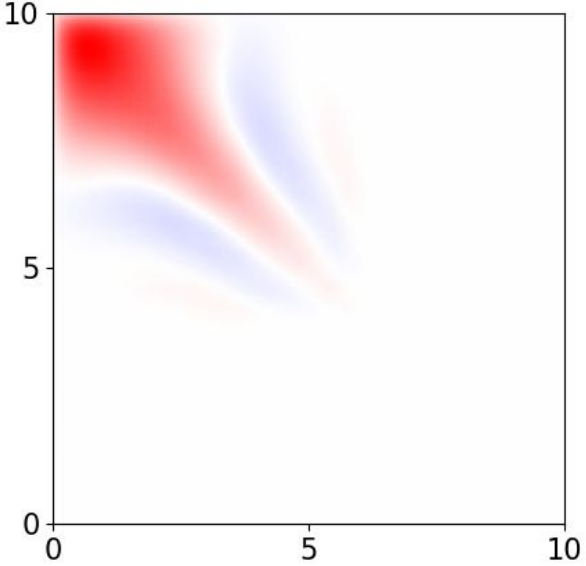


Simulation results

Monte Carlo Wavefunction

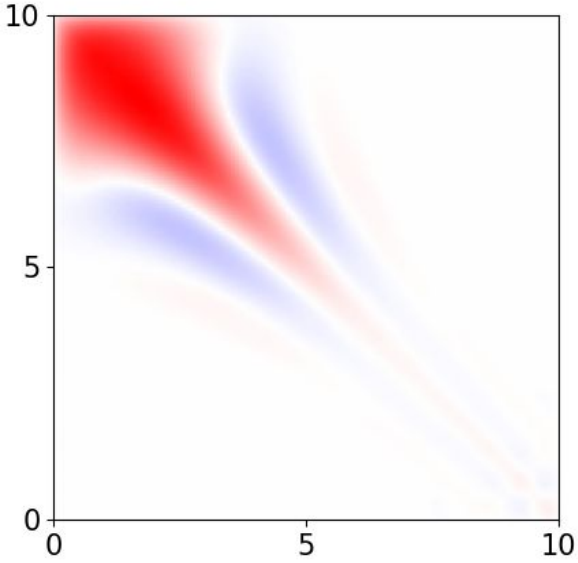


Neural Network

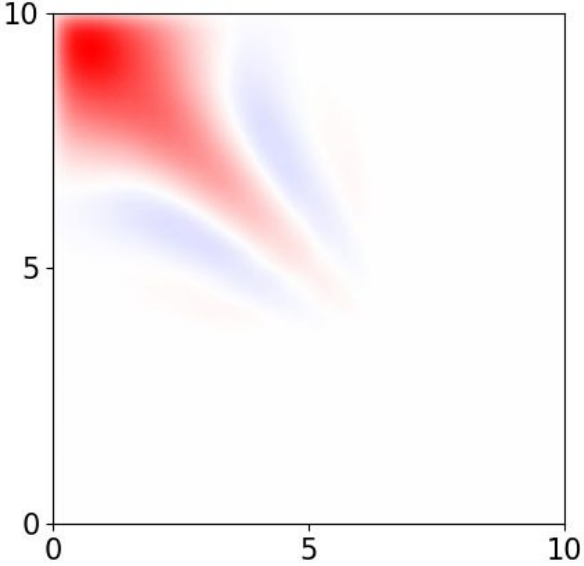


Simulation results

Monte Carlo Wavefunction

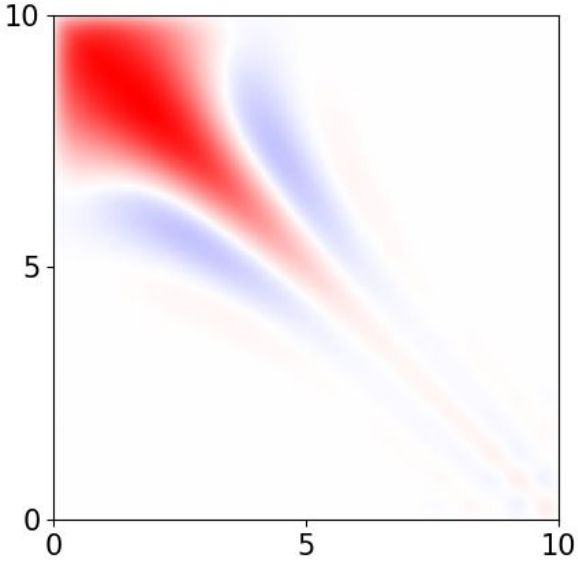


Neural Network

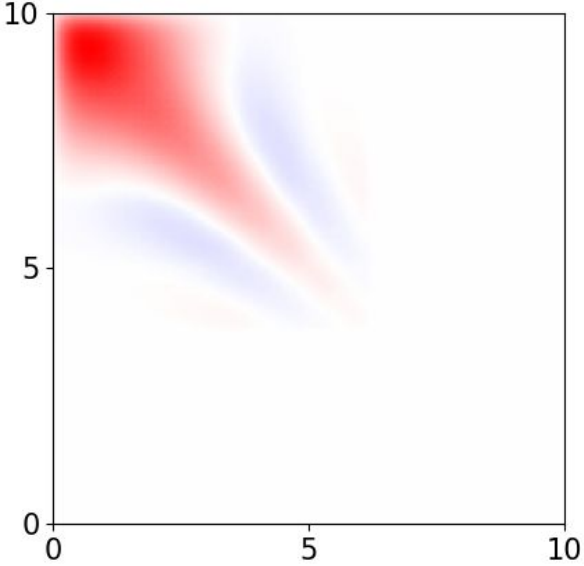


Simulation results

Monte Carlo Wavefunction

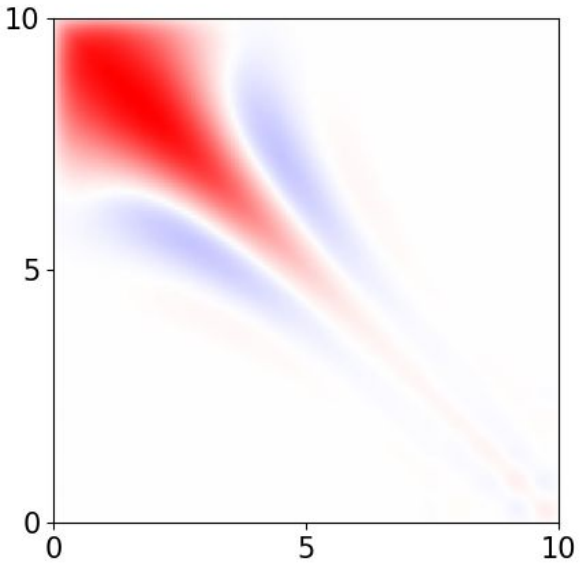


Neural Network

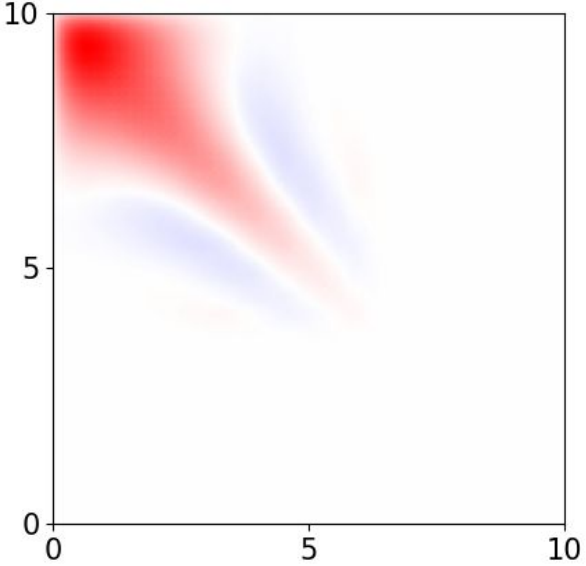


Simulation results

Monte Carlo Wavefunction

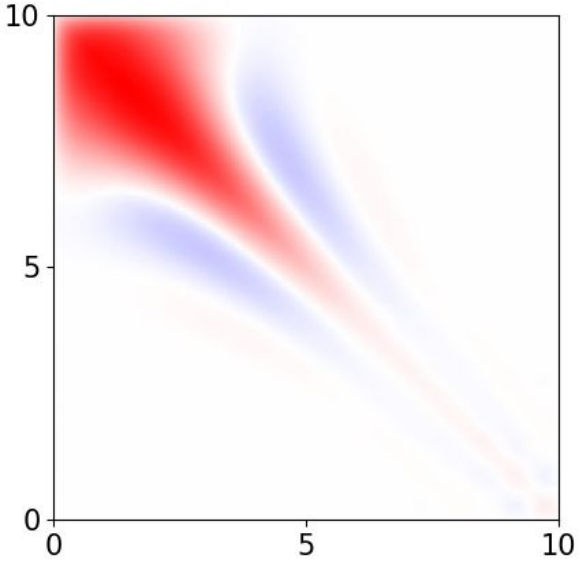


Neural Network

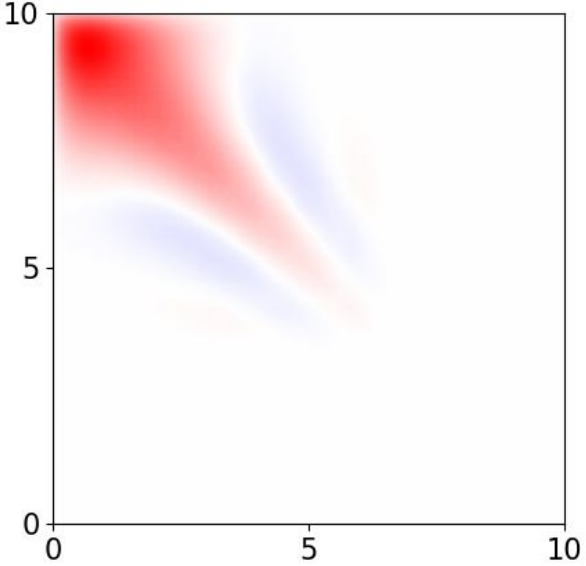


Simulation results

Monte Carlo Wavefunction

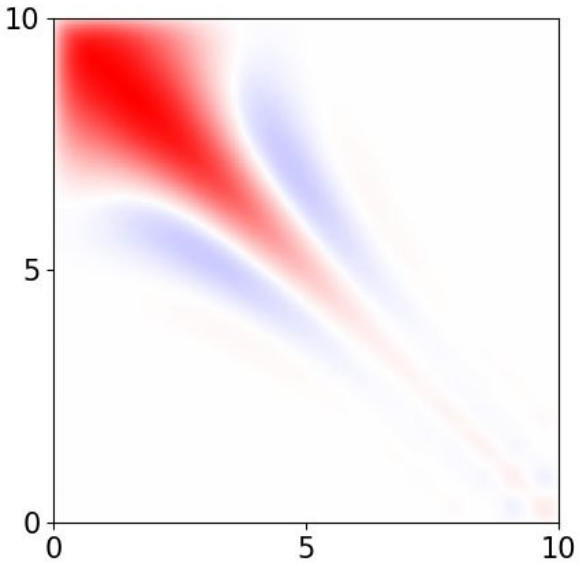


Neural Network

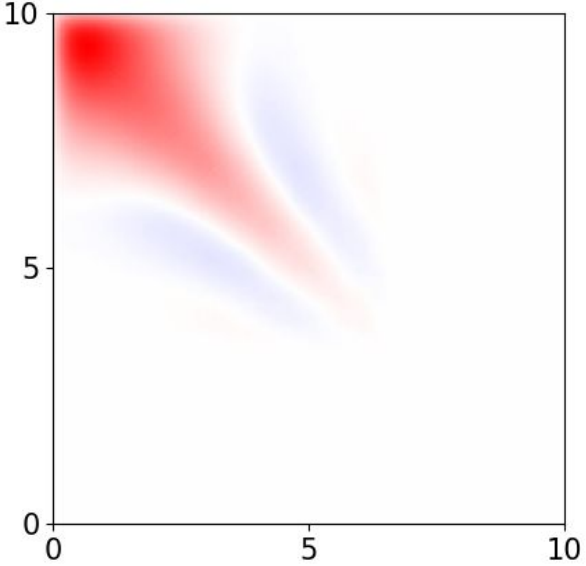


Simulation results

Monte Carlo Wavefunction

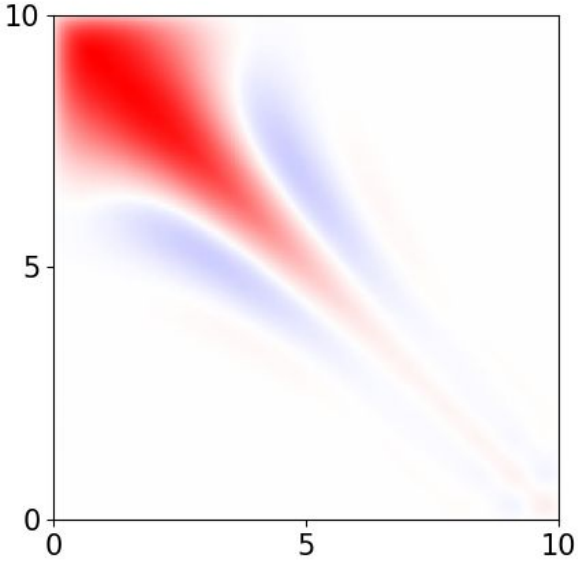


Neural Network

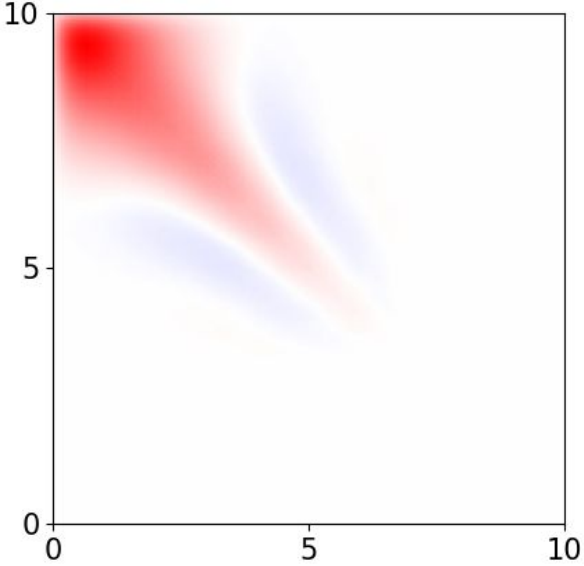


Simulation results

Monte Carlo Wavefunction

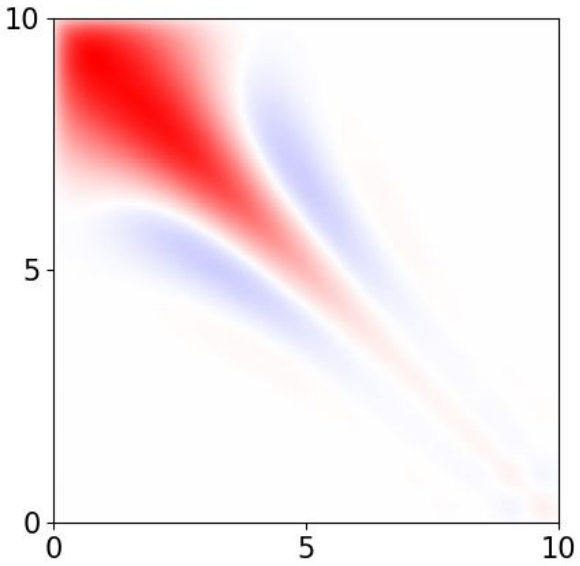


Neural Network

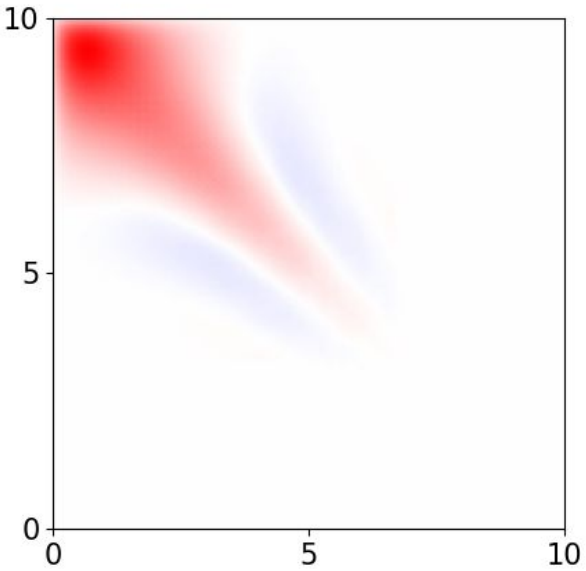


Simulation results

Monte Carlo Wavefunction

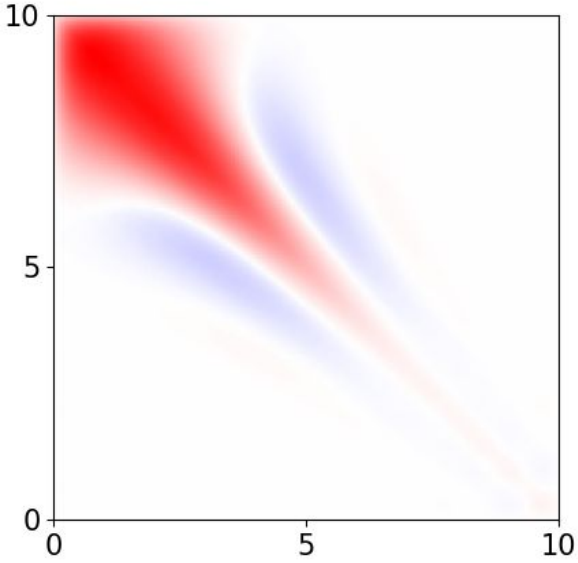


Neural Network

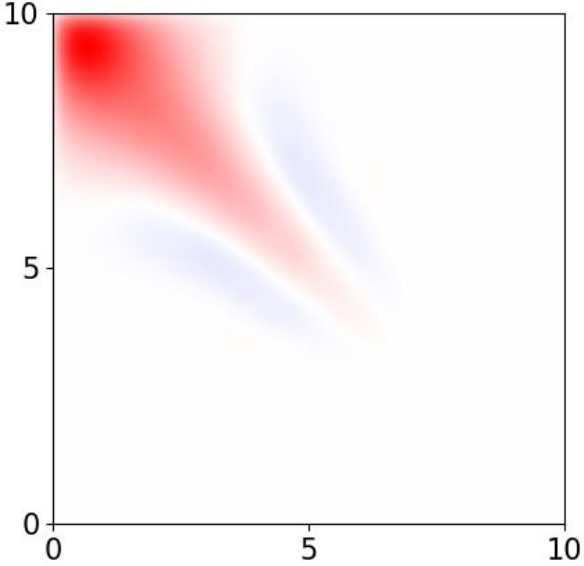


Simulation results

Monte Carlo Wavefunction

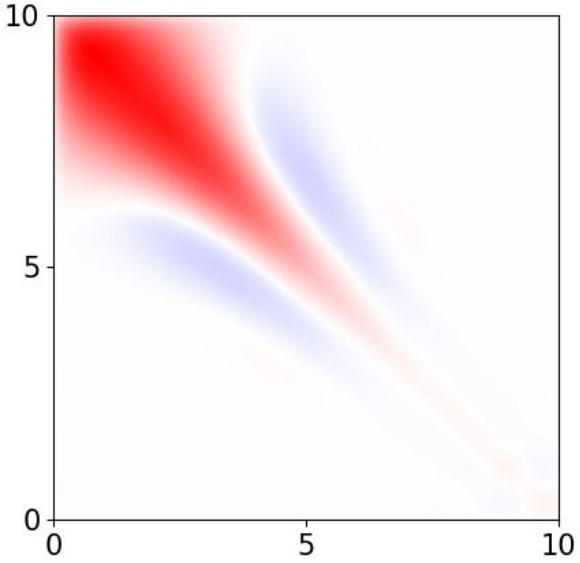


Neural Network

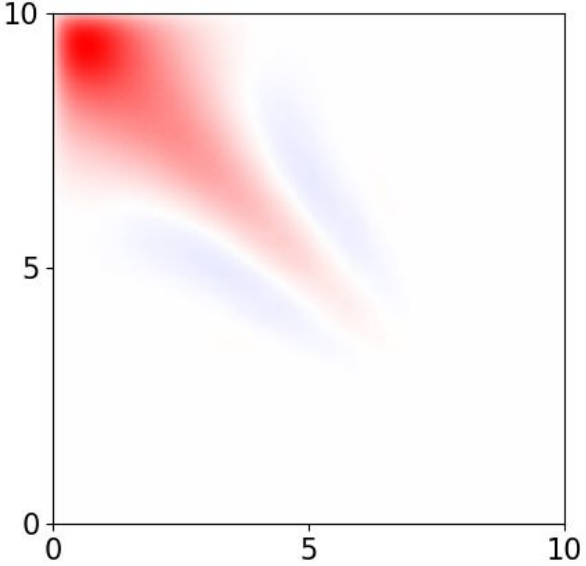


Simulation results

Monte Carlo Wavefunction

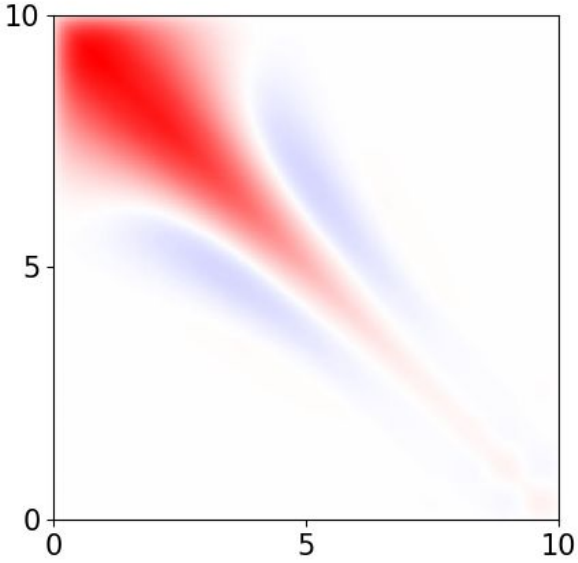


Neural Network

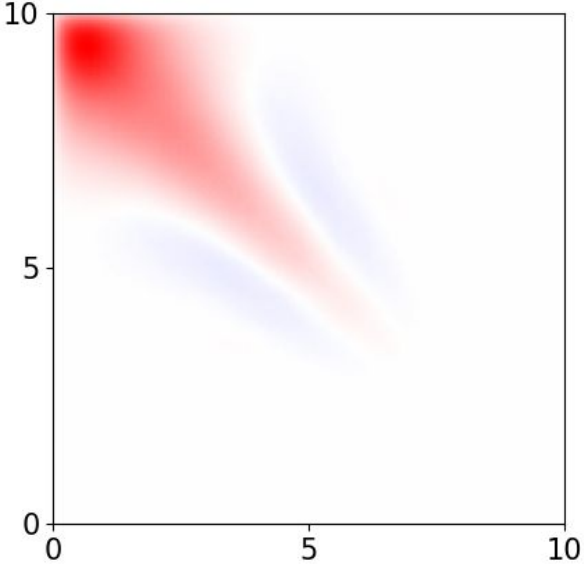


Simulation results

Monte Carlo Wavefunction

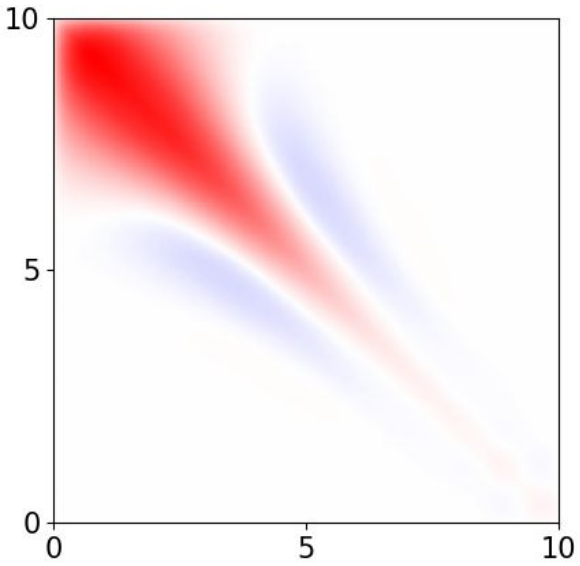


Neural Network

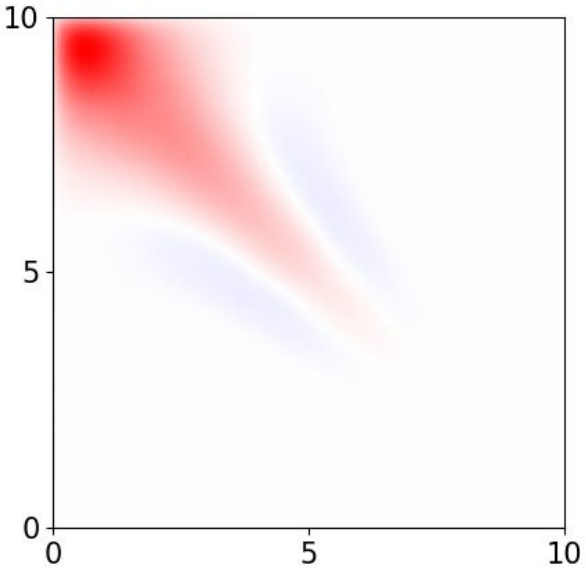


Simulation results

Monte Carlo Wavefunction

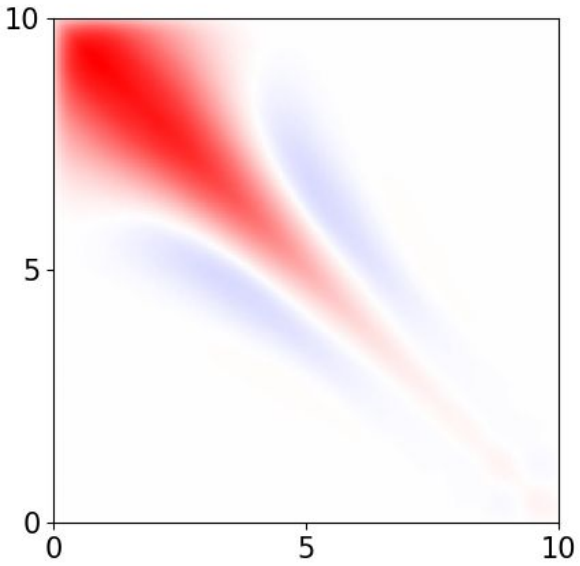


Neural Network

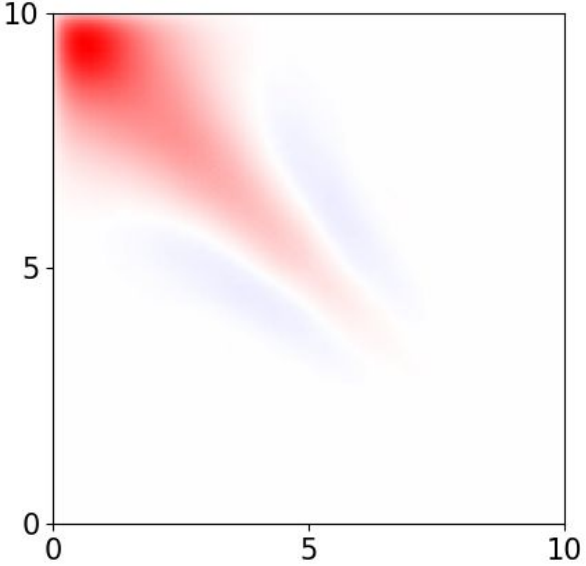


Simulation results

Monte Carlo Wavefunction

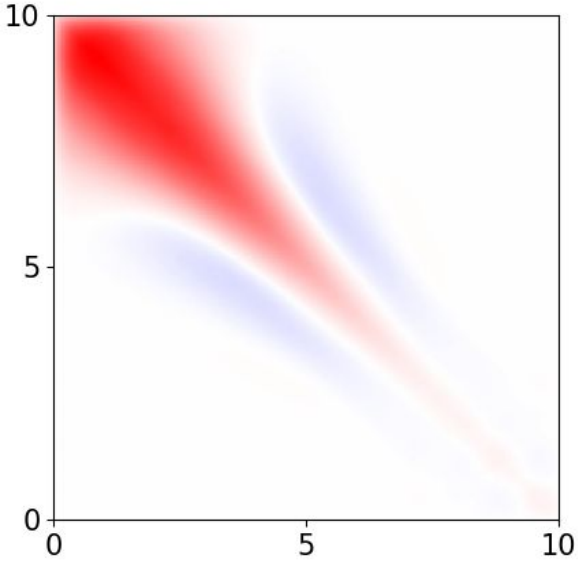


Neural Network

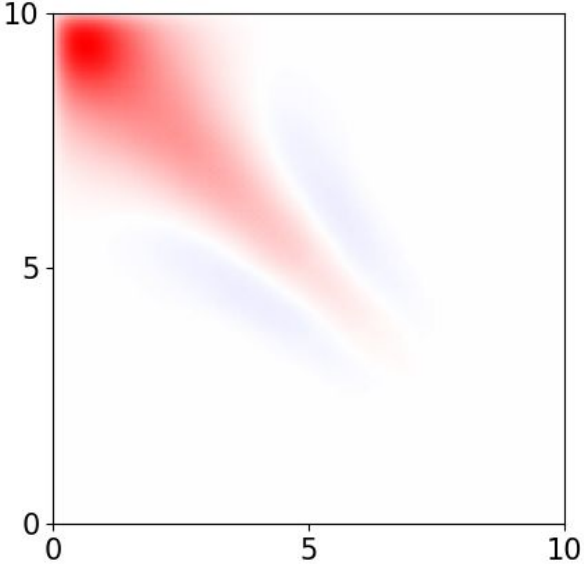


Simulation results

Monte Carlo Wavefunction

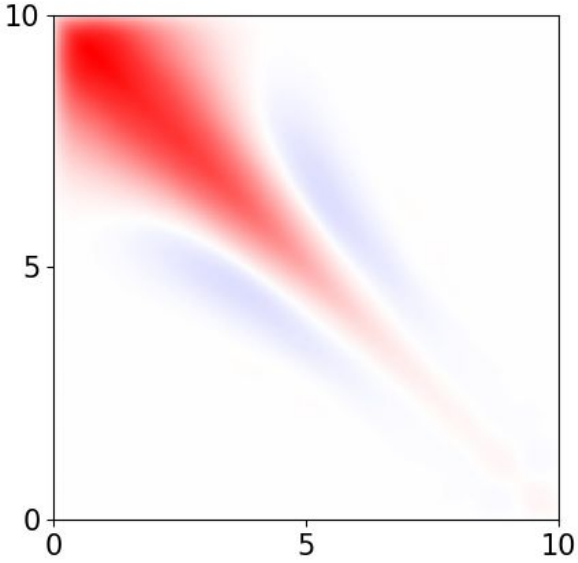


Neural Network

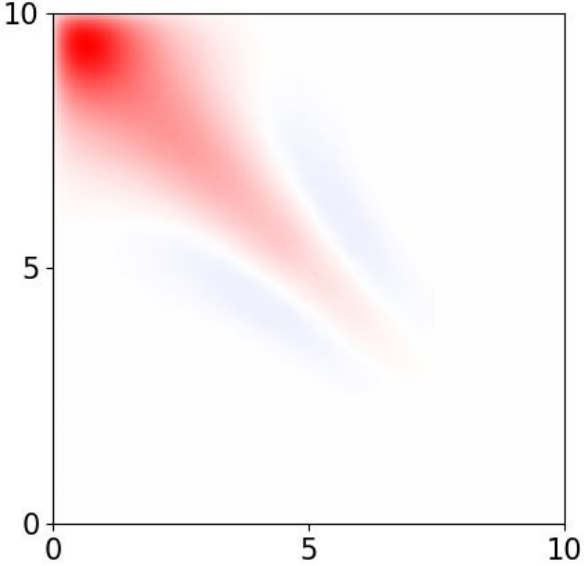


Simulation results

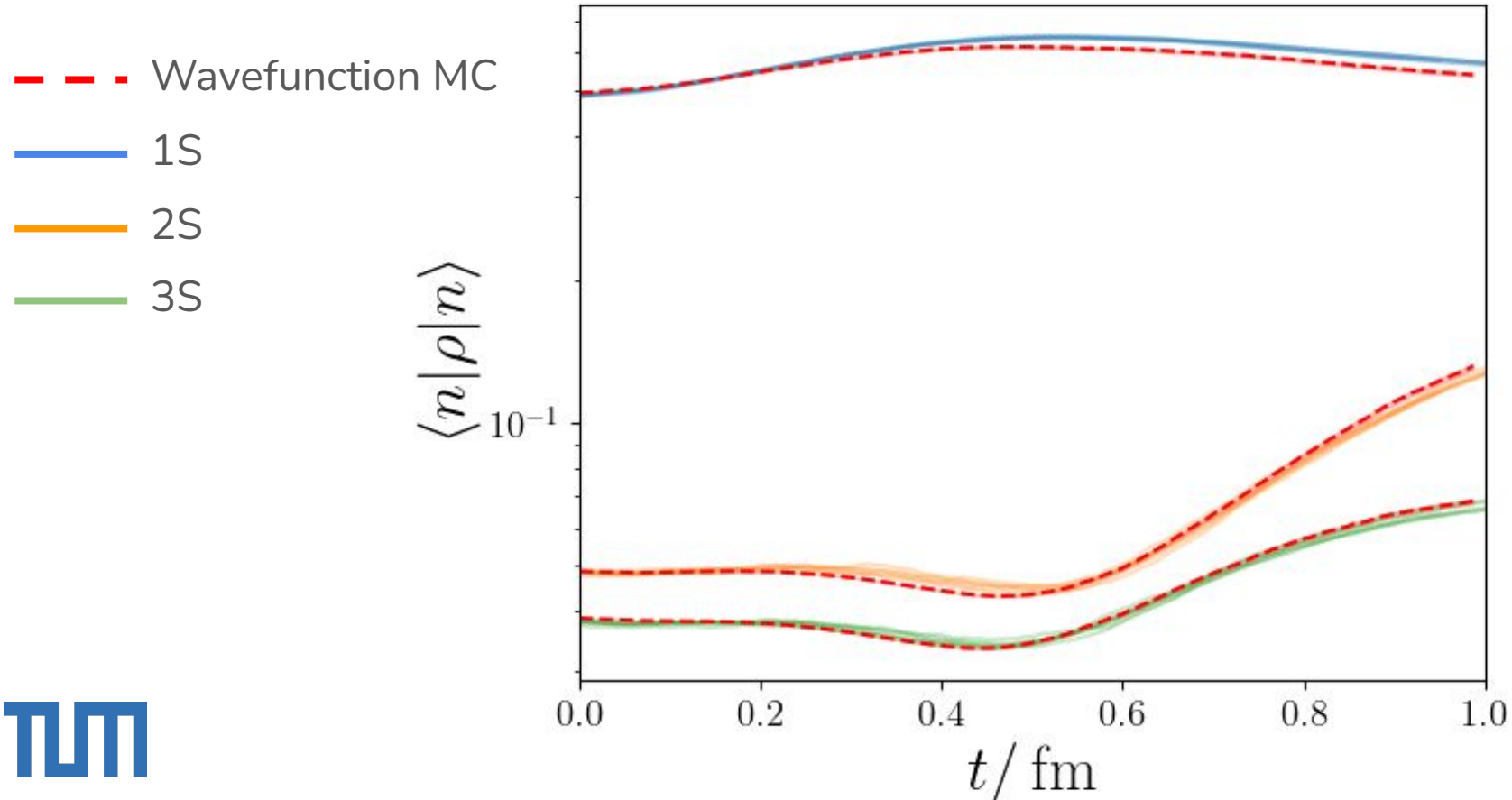
Monte Carlo Wavefunction



Neural Network



Simulation results



✓

- Time evolution works
- Scaling to larger systems seems to work
- Reasonable stability around certain parameters spots

?

- Optimization still seems very hard ? (use something geometry aware?)
- How far can we scale ?

Outlook

How far is this scalable ?

larger hilbert spaces ?

Scalar field theory?

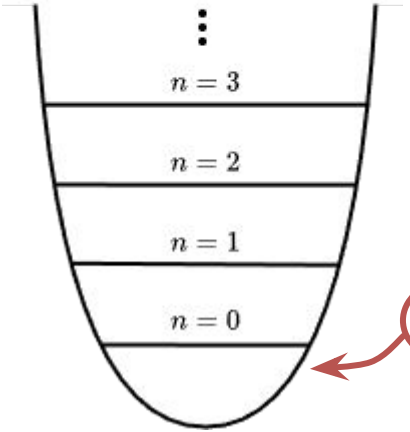
Many body dynamics?

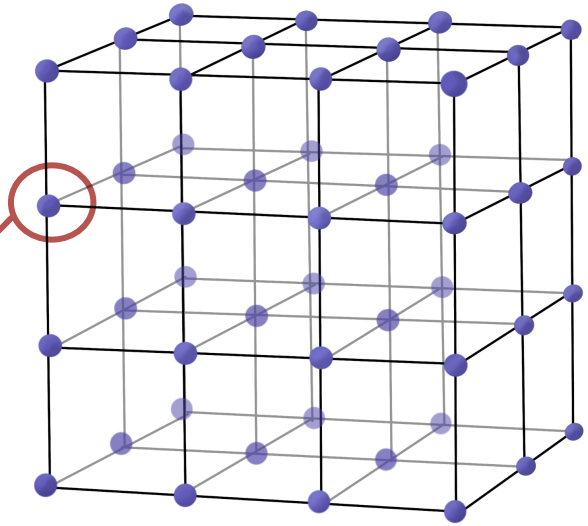
Gauge theories?

- **Guarantees of correctness**
- **What do we really need for scale ?**

Backup

Hamiltonian methods





$$|\phi\rangle \in \mathcal{H}$$

$$i\frac{d}{dt}|\phi(t)\rangle = H|\phi(t)\rangle$$

Caldeira leggett

$$\frac{d\rho}{dt} = -i[H,\rho] + C\rho C^\dagger - \frac{1}{2}\{C^\dagger C, \rho\}$$

$$H = \frac{p^2}{M} - \frac{\alpha}{x} + \beta\{x,p\}$$

$$C = x + \gamma i p$$

Time evolution: p-tVMC Control variate

- Instead of the local infidelity estimator

$$\mathcal{I}_{\mathrm{loc}}(\sigma,\eta) = 1 - \frac{\langle\sigma|U|\phi\rangle\,\langle\eta|U^\dagger|\psi\rangle}{\langle\sigma|\psi\rangle\,\langle\eta|\phi\rangle}$$

- We can subtract a Control Variate

$$\mathcal{I}_{\mathrm{loc}}^{CV} = \mathrm{Re}\mathcal{I}_{\mathrm{loc}} - c(|1-\mathcal{I}_{\mathrm{loc}}|^2) - 1).$$

- Want to improve the signal to noise ratio when $\mathcal{I}_{\mathrm{loc}} \to 0$
- In this limit one can proof the optimal $c = -\frac{1}{2}$

Cranck Nicholoson

$$\left(1 - \frac{dt}{2}L\right)\psi_\Theta(t+dt) = \left(1 + \frac{dt}{2}L\right)\psi(t),$$

$$U_- = 1 - \frac{dt}{2}L$$

$$U_+ = 1 + \frac{dt}{2}L$$

$$F_{loc} = \frac{\langle\sigma|U_+|\phi\rangle\,\langle\eta|U_-|\psi_\Theta\rangle}{\langle\sigma|U_-|\psi_\Theta\rangle\,\langle\eta|U_+|\phi\rangle}$$