Title: Quantum and Classical Dynamics from a Time Dependent Variational Principle

Speakers:

Collection: Machine Learning for Quantum Many-Body Systems

Date: June 13, 2023 - 2:45 PM

URL: https://pirsa.org/23060036

Time-dependent variational principle for quantum and classical dynamics

Perimeter Institute, 06/13/23

Moritz Reh[1], Markus Schmitt[2], Martin Gärttner[1,3,4]

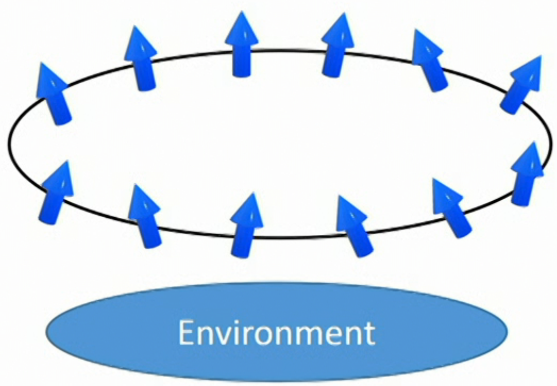
[1]Kirchhoff-Institut für Physik, Universität Heidelberg, Im Neuenheimer Feld 227, 69120 Heidelberg, Germany
[2]Institut für Physik, Universität Regensburg, Universitätsstrasse 31,
93053 Regensburg, Germany
[3]Physikalisches Institut, Universität Heidelberg, Im Neuenheimer Feld 226, 69120 Heidelberg, Germany
[4]Institut für Theoretische Physik, Ruprecht-Karls-Universität Heidelberg, Philosophenweg 16, 69120 Heidelberg, Germany

Dynamics of Open Quantum Systems

Why simulate Open Quantum Systems?

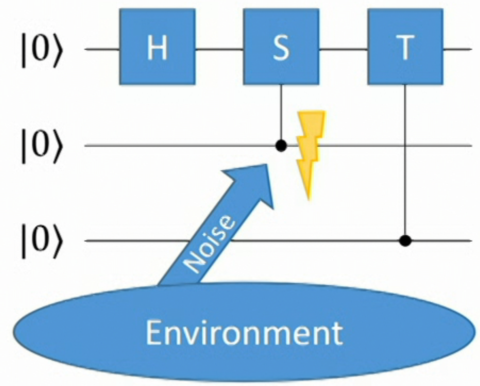
..to discover new physics:

A lack of computational tools prevents the exploration of new interesting physics:



..as a benchmarking tool:

Quantum simulators are sensitive to outside noise – require tools to benchmark these devices:



Evolution equation of the quantum state ρ:

$$\dot{\rho} = -i[H,\rho] + \gamma \sum_i L^i \rho L^{i\dagger} - \frac{1}{2}\{L^{i\dagger} L^i, \rho\}$$

with the spin-Hamiltonian

$$H = \Sigma_{d\in\{x,y,z\}} \Sigma_{\langle ij\rangle} J^d \sigma_i^d \sigma_j^d + h^d \sigma_i^d$$

POVM representations

Mapping density matrices to probability distributions

Quantum state $\hat{\rho}$

$$\begin{pmatrix} & \cdots & \\ \vdots & \ddots & \vdots \\ & \cdots & \end{pmatrix}$$

$$P^{\boldsymbol{a}} = \mathrm{tr}(\widehat{M}^{\boldsymbol{a}}\hat{\rho})$$

$$\hat{\rho} = \mathrm{tr}(P^{\boldsymbol{a}}T^{-1\boldsymbol{a}\boldsymbol{a}'}\widehat{M}^{\boldsymbol{a}'})$$

POVM-distribution $P^{\boldsymbol{a}}$

POVM representations

Mapping density matrices to probability distributions

Quantum state $\hat{\rho}$

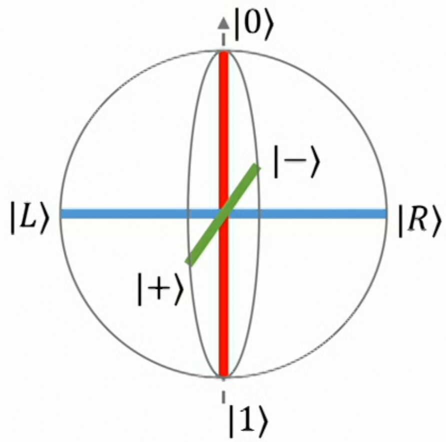
$$\begin{pmatrix} & \cdots & \\ \vdots & \ddots & \vdots \\ & \cdots & \end{pmatrix}$$

$$P^{\boldsymbol{a}} = \mathrm{tr}(\widehat{M}^{\boldsymbol{a}}\hat{\rho})$$

$$\hat{\rho} = \mathrm{tr}(P^{\boldsymbol{a}}T^{-1\boldsymbol{a}\boldsymbol{a}'}\widehat{M}^{\boldsymbol{a}'})$$

POVM-distribution $P^{\boldsymbol{a}}$

Measurement Operator:



POVM representations

Mapping density matrices to probability distributions

Quantum state $\hat{\rho}$

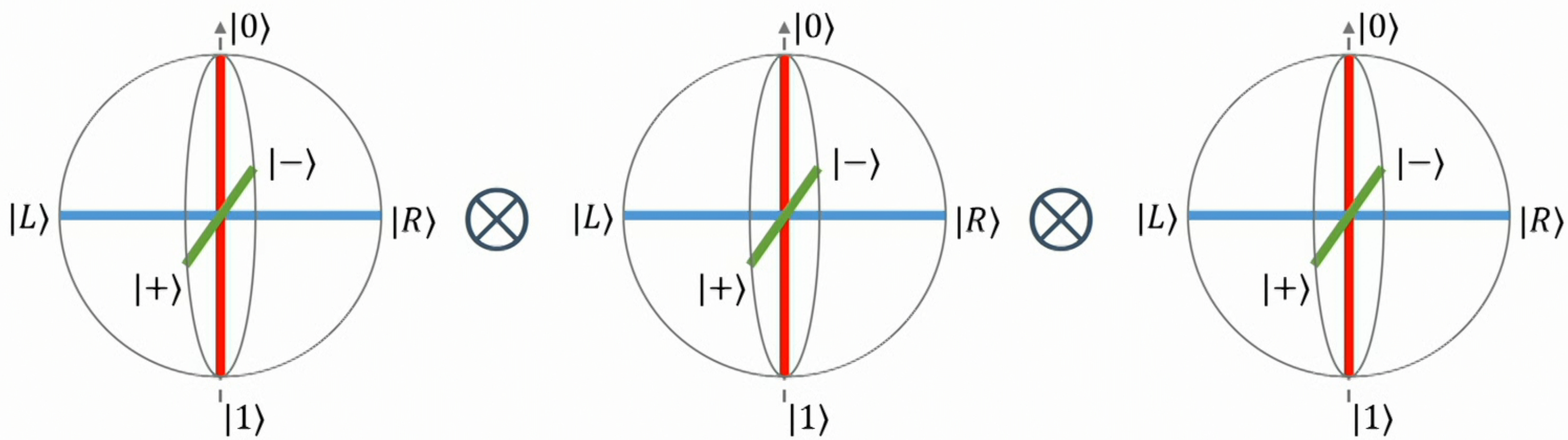
$$\begin{pmatrix} & \cdots & \\ \vdots & \ddots & \vdots \\ & \cdots & \end{pmatrix}$$

$$P^{\boldsymbol{a}} = \mathrm{tr}(\hat{M}^{\boldsymbol{a}}\hat{\rho})$$

$$\hat{\rho} = \mathrm{tr}(P^{\boldsymbol{a}} T^{-1\boldsymbol{a}\boldsymbol{a}'} \hat{M}^{\boldsymbol{a}'})$$

POVM-distribution $P^{\boldsymbol{a}}$

Measurement Operator:



Neural Network State Encodings

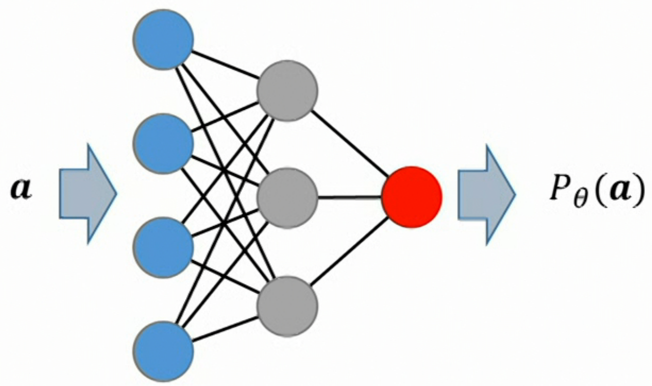
Defeating the curse of dimensionality

Idea: Instead of storing the entire distribution, store a **function that approximates it**!

Neural Network State Encodings

Defeating the curse of dimensionality

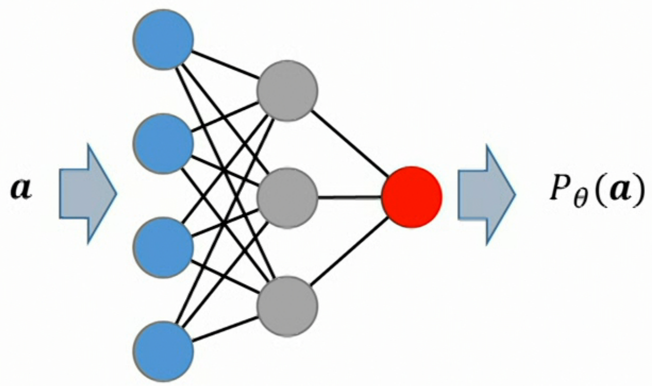
Idea: Instead of storing the entire distribution, store a **function that approximates it**!



Neural Network State Encodings

Defeating the curse of dimensionality

Idea: Instead of storing the entire distribution, store a **function that approximates it**!

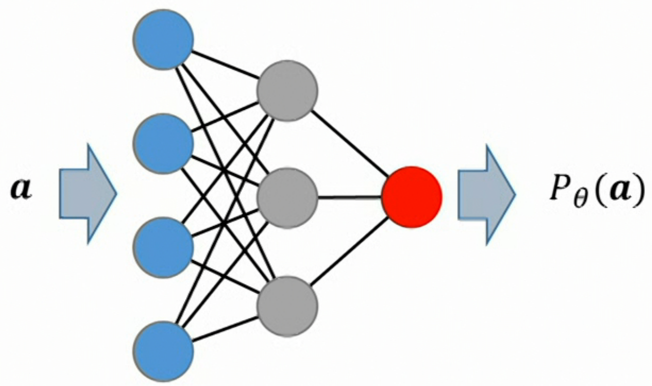


Questions:
What is a clever encoding of the distribution?
How to track the evolution of a quantum state?

Neural Network State Encodings

Defeating the curse of dimensionality

Idea: Instead of storing the entire distribution, store a **function that approximates it**!

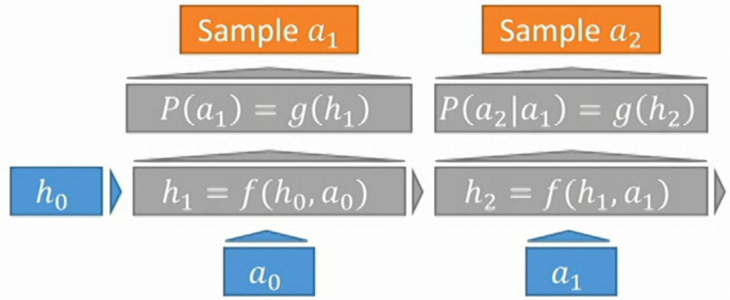


Questions:

What is a clever encoding of the distribution?

How to track the evolution of a quantum state?

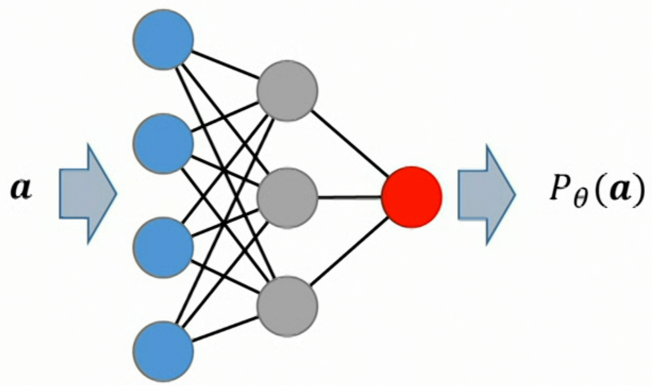
Architecture choice: Recurrent Neural Networks (RNNs)



Neural Network State Encodings

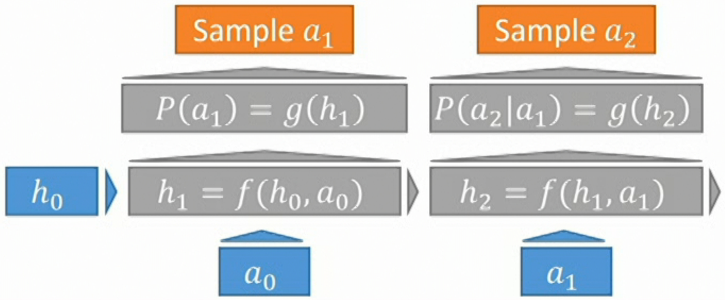
Defeating the curse of dimensionality

Idea: Instead of storing the entire distribution, store a **function that approximates it**!



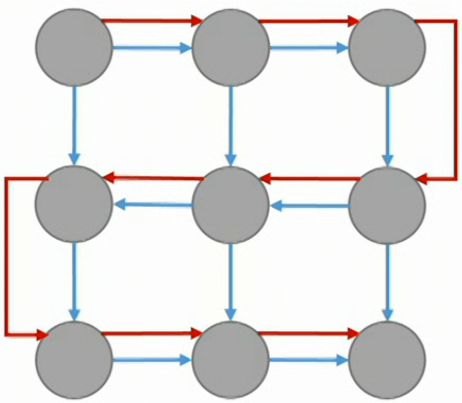
Questions:
What is a clever encoding of the distribution?
How to track the evolution of a quantum state?

Architecture choice: Recurrent Neural Networks (RNNs)



In 2D:
No naive chains!
(Hibat-Allah et. al., Phys. Rev. Research 2, 023358)

Red: Sampling path
Blue: Hidden states



Time-evolution algorithm

How to recover the time-evolution for the network parameters

Network Initialization	Time Evolution	Observables
Network parameters are analytically set to product states at $t = 0$	At each time t the TDVP Eq. is solved using generated samples	Observables are estimated from the samples in each time step

Time-evolution algorithm

How to recover the time-evolution for the network parameters

Network Initialization	Time Evolution	Observables
Network parameters are analytically set to product states at $t = 0$	At each time t the TDVP Eq. is solved using generated samples	Observables are estimated from the samples in each time step

The Time-Dependent Variational Principle (TDVP)

Setting: At time t the network parameters $\theta(t)$ encode the probability distribution $P_{\theta(t)}$. For $P_{\theta(t)}$ we know the time derivative, i.e.

$$\dot{P}^{\boldsymbol{a}}_{\theta(t)} = \mathcal{L}^{\boldsymbol{ab}} P^{\boldsymbol{b}}_{\theta(t)}.$$

How do we obtain the corresponding parameter update $\dot{\theta}(t)$?

Time-evolution algorithm

How to recover the time-evolution for the network parameters

Network Initialization

Network parameters are analytically set to product states at $t = 0$

Time Evolution

At each time t the TDVP Eq. is solved using generated samples

Observables

Observables are estimated from the samples in each time step

The Time-Dependent Variational Principle (TDVP)

Setting: At time t the network parameters $\theta(t)$ encode the probability distribution $P_{\theta(t)}$. For $P_{\theta(t)}$ we know the time derivative, i.e.

$$\dot{P}^{\boldsymbol{a}}_{\theta(t)} = \mathcal{L}^{\boldsymbol{ab}} P^{\boldsymbol{b}}_{\theta(t)}.$$

How do we obtain the corresponding parameter update $\dot{\theta}(t)$?

Idea: Minimize a distance measure between the forward propagated distribution $P^{\boldsymbol{a}}_{\theta(t)} + \tau \dot{P}^{\boldsymbol{a}}_{\theta(t)}$ and a *trial* evolution from the network $P^{\boldsymbol{a}}_{\theta(t)+\tau\dot{\theta}(t)}$.

Minimize

$$\mathcal{D}(P^{\boldsymbol{a}}_{\theta(t)+\tau\dot{\theta}(t)}, P^{\boldsymbol{a}}_{\theta(t)} + \tau \dot{P}^{\boldsymbol{a}}_{\theta(t)})$$

w.r.t. $\dot{\theta}$.

Obtain:

$$S_{kk'}\dot{\theta}_{k'} = F_k$$

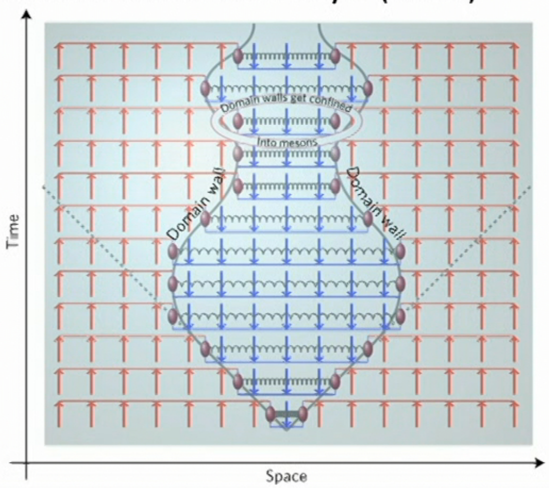
Notes:

- $$S_{kk'} = \left\langle \frac{\partial \log P^{\boldsymbol{a}}}{\partial \theta_k} \frac{\partial \log P^{\boldsymbol{a}}}{\partial \theta_{k'}} \right\rangle_{\boldsymbol{a}\sim P}$$ $$F_k = \left\langle \frac{\partial \log P^{\boldsymbol{a}}}{\partial \theta_k} \mathcal{L}^{\boldsymbol{ab}} \frac{P^{\boldsymbol{b}}}{P^{\boldsymbol{a}}} \right\rangle_{\boldsymbol{a}\sim P}$$ are obtainable through a sub-exponential number of samples (computationally intense part!)
- Not all distance measures are equally suited for this task, e.g. the L_2 norm doesn't allow for sampling

Confinement Physics

What effect does dissipation have on the confinement dynamics?

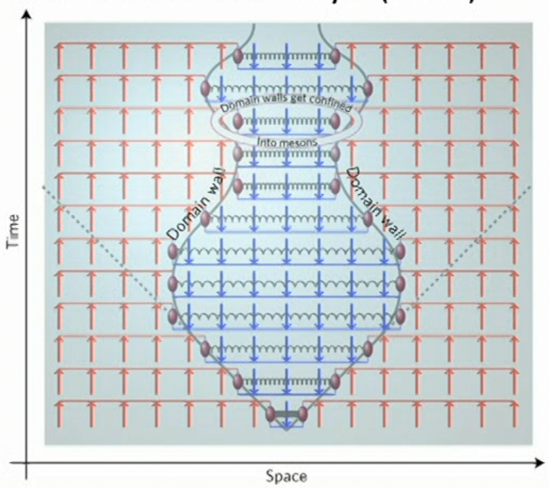
Originial work: Real-time confinement following a quantum quench to a non-integrable model, Kormos et. al. Nat. Phys. (2017)



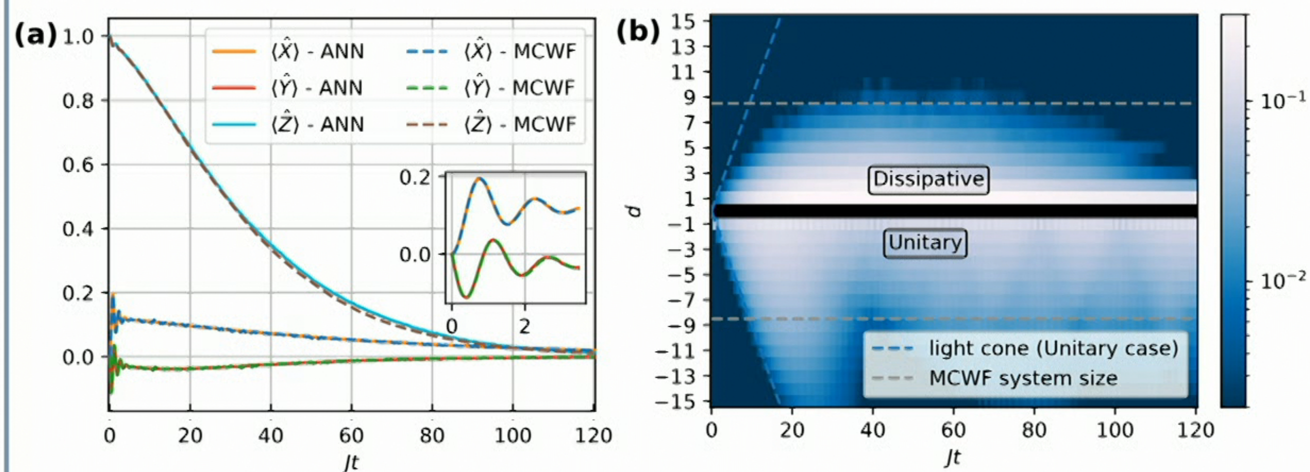
Confinement Physics

What effect does dissipation have on the confinement dynamics?

Originial work: Real-time confinement following a quantum quench to a non-integrable model, Kormos et. al. Nat. Phys. (2017)



Augmented study: Spin-chain of length $L = 32$ with nearest neighbor couplings $H = \Sigma_i \, \sigma_i^z \sigma_{i+1}^z + h^z \sigma_i^z + h^x \sigma_i^x$ with $h^z = 0.05$, $h^x = 0.25$ and $\gamma = 0.25$, $L = \sigma^z$

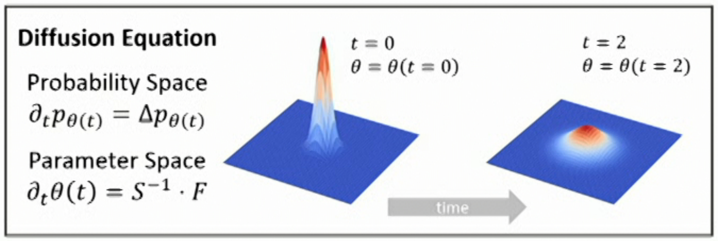


Extension to partial differential equations

Adapt formalism to Fokker-Planck type equations

Idea: Minimization of

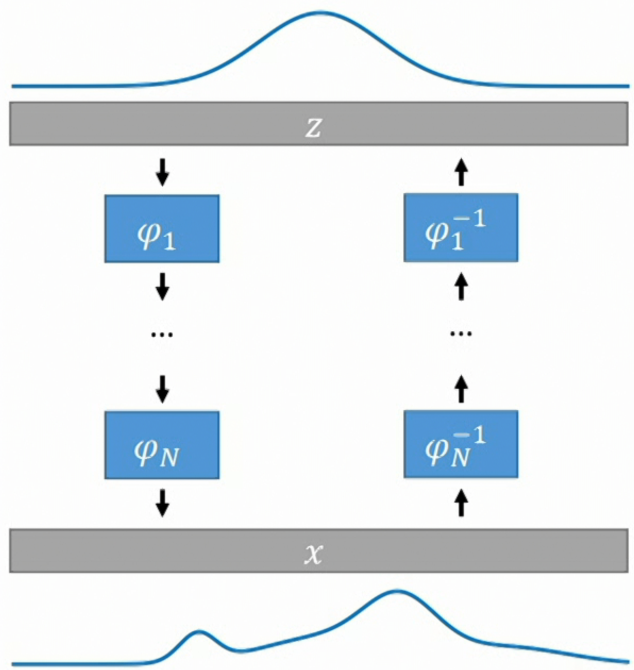
$$\mathcal{D}\left(p_{\theta(t)+\tau\dot{\theta}(t)}(\boldsymbol{x}), p_{\theta(t)}(\boldsymbol{x}) + \tau\dot{p}_{\theta(t)}(\boldsymbol{x})\right)$$



Necessary adaptations:

- Replace network to approximate continuous distributions
- Replace evolution equation by partial differential equation

Architecture Details: Utilize Normalizing Flows (NF), to obtain a parameterized ansatz function.



Set of coupling blocks $\{\boldsymbol{\varphi}\}$:

Encode coordinate transform from latent space to real space

Real space probability obtained via change of coordinate transform:

$$p(x) = p_L(z) \cdot \left|\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{x}}\right|$$

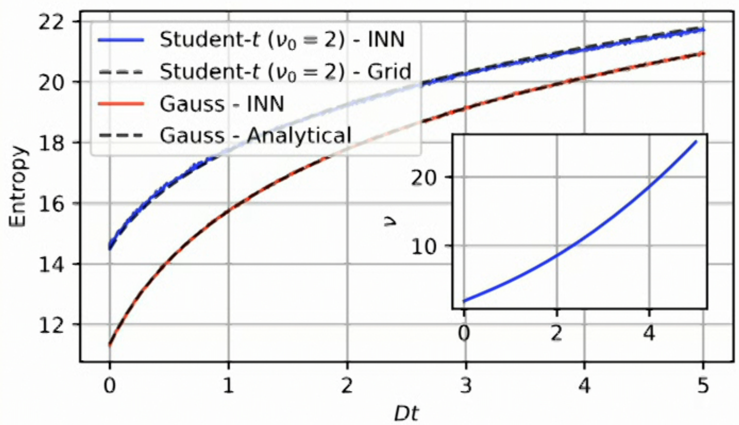
Extension to partial differential equations

Adapt formalism to Fokker-Planck type equations

Application: Heat Equation in eight dimensions

$$\partial_t p(t, \boldsymbol{x}) = D\Delta_{\mathbf{x}} \mathrm{p}(\mathrm{t}, \mathbf{x}),$$

with $p(0, \boldsymbol{x}) \propto \left(1 + \frac{\boldsymbol{x}^2}{\nu}\right)^{-(\nu+d)/2}$



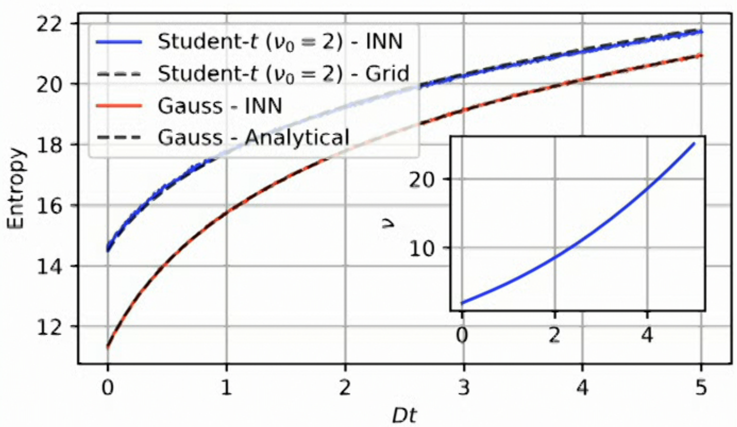
Extension to partial differential equations

Adapt formalism to Fokker-Planck type equations

Application: Heat Equation in eight dimensions

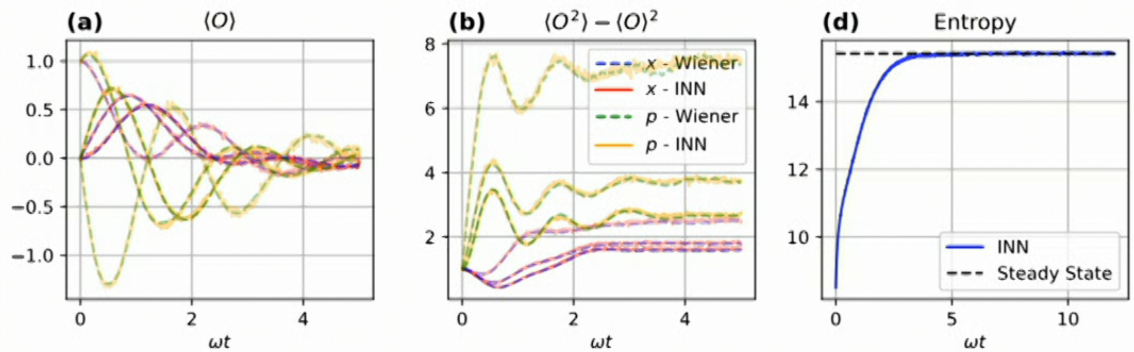
$$\partial_t p(t, \boldsymbol{x}) = D\Delta_{\mathbf{x}} \mathrm{p}(\mathrm{t}, \mathbf{x}),$$

with $p(0, \boldsymbol{x}) \propto \left(1 + \frac{\boldsymbol{x}^2}{\nu}\right)^{-(\nu+d)/2}$



Application: Continuous phase-space dynamics of 3 coupled oscillators with dissipation to a thermal bath (6-dimensional phase space).

EOM: $$\partial_t \rho(t, \mathbf{x}, \mathbf{p}) = [-\partial_{\mathbf{p}} H \cdot \partial_{\mathbf{x}} + \partial_{\mathbf{x}} H \cdot \partial_{\mathbf{p}} + \gamma \left(\mathbf{p} \cdot \partial_{\mathbf{p}} + m k_B \textstyle\sum_i T_i \partial^2_{p_i}\right)] \, \rho(t, \mathbf{x}, \mathbf{p})$$



Evolution of the means (a), variances (b) and entropy (c). The system's initial configuration is $\boldsymbol{x} = (1, 0, 0)$ and $\boldsymbol{p} = (0, 1, 0)$, with temperatures $k_b \boldsymbol{T}/m\omega^2 = (10, 3, 1)$.

Outlook

Problems & Remarks

Further Questions:

- What are the fundamental restrictions on neural quantum states?
- Optimally suited network architecture? (Also see: **PRB 107**, 195115)
- ...

Outlook

Problems & Remarks

Further Questions:

- What are the fundamental restrictions on neural quantum states?
- Optimally suited network architecture? (Also see: **PRB 107**, 195115)
- ...

Further Reading:

- Open Quantum Dynamics with NQS: **Phys. Rev. Lett. 127**, 230501
- TDVP for PDEs: **Mach. Learn.: Sci. Technol. 3**, 04LT02

Thanks to my collaborators!

Martin Gärttner, Heidelberg University

Markus Schmitt, Regensburg University

Outlook

Problems & Remarks

Further Questions:

- What are the fundamental restrictions on neural quantum states?
- Optimally suited network architecture? (Also see: **PRB 107**, 195115)
- ...

Further Reading:

- Open Quantum Dynamics with NQS: **Phys. Rev. Lett. 127**, 230501
- TDVP for PDEs: **Mach. Learn.: Sci. Technol. 3**, 04LT02

Thanks to my collaborators!

Martin Gärttner,
Heidelberg University

Markus Schmitt,
Regensburg University

Questions?