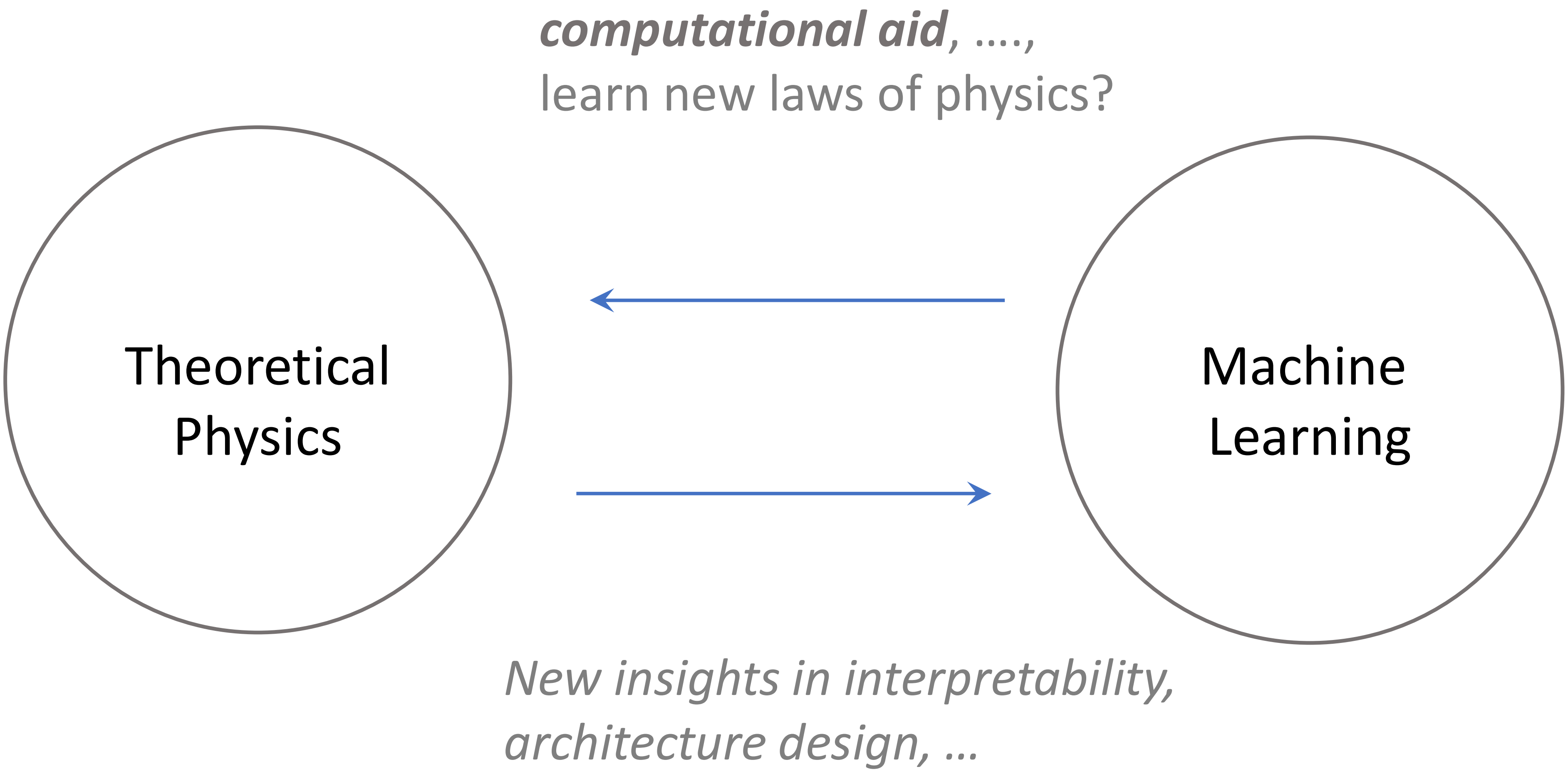
Machine Learning for Computing in Quantum Field Theory

Miranda C. N. Cheng

University of Amsterdam, the Netherlands
and
Academia Sinica, Taipei, Taiwan



computational aid, ….,
learn new laws of physics?
Theoretical
Physics
Machine
Learning
New insights in interpretability,
architecture design, …

Based on

Scaling Up Machine Learning For Quantum Field Theory with Equivariant Continuous Flows

ArXiv: 2110.02673[cs.LG]

NeurIPS 2021 ML4Physical Sciences Workshop



Pim de Haan
QualComm AI Research

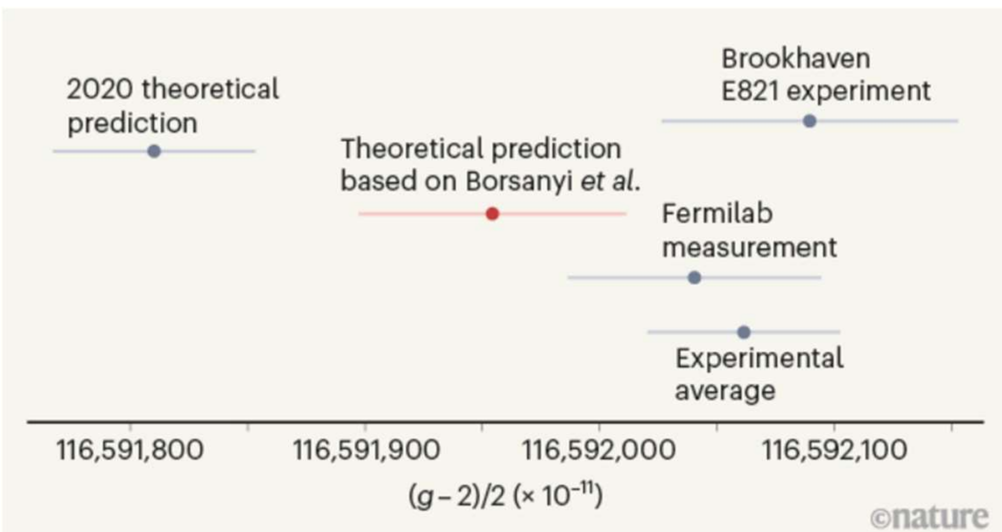
Corrado Rainone
QualComm AI Research

Roberto Bondesan
QualComm AI Research

Motivation

Physicists would like to disentangle new physics (new particles, dark matter, …) from data, gathered from the detectors or from the sky. For that we need very high-precision predictions from known physics, such as the SM. We cannot reliably do that right now.

e.g. muon magnetic moment



Nature, 2021

'Last Hope' Experiment Finds Evidence for Unknown Particles

Motivation

Physicists would like to disentangle new physics (new particle, dark matter, …) from data, gathered from the detectors or from the sky. For that we need very high-precision predictions from known physics, such as the SM. We cannot do that right now.

Question: Can ML help?

Motivation

- Lattice field theory has been the main computation tool for computing in non-perturbative QFT, such as QCD, computations.

e.g. a single scalar field

$$\phi : \mathbb{R}^D \to \mathbb{R} \ni \phi(x) \rightsquigarrow \phi : (\mathbb{Z}/L\mathbb{Z})^{\otimes D} \to \mathbb{R} \ni \phi_x$$

$$\langle \mathcal{O} \rangle = \int \mathcal{D}\phi e^{-S[\phi(x)]} \mathcal{O}(\phi(x)) \rightsquigarrow \langle \mathcal{O} \rangle = \frac{1}{N_{\mathrm{conf}}} \sum_{i=1}^{N_{\mathrm{conf}}} \mathcal{O}(\phi_x^{(i)})$$

sampling,
usually done by Markov Chain Monte Carlo (MCMC)

Motivation

$$\langle \mathcal{O} \rangle = \int \mathcal{D}\phi e^{-S[\phi(x)]}\, \mathcal{O}(\phi(x)) \rightsquigarrow \langle \mathcal{O} \rangle = \frac{1}{N_{\mathrm{conf}}} \sum_{i=1}^{N_{\mathrm{conf}}} \mathcal{O}(\phi_x^{(i)})$$

sampling,
usually done by Markov Chain Monte Carlo (MCMC)

- Challenge: **critical slowing down,** or the catastrophic inefficiency which makes it taking prohibitively long to sample enough independent configurations in the limits of

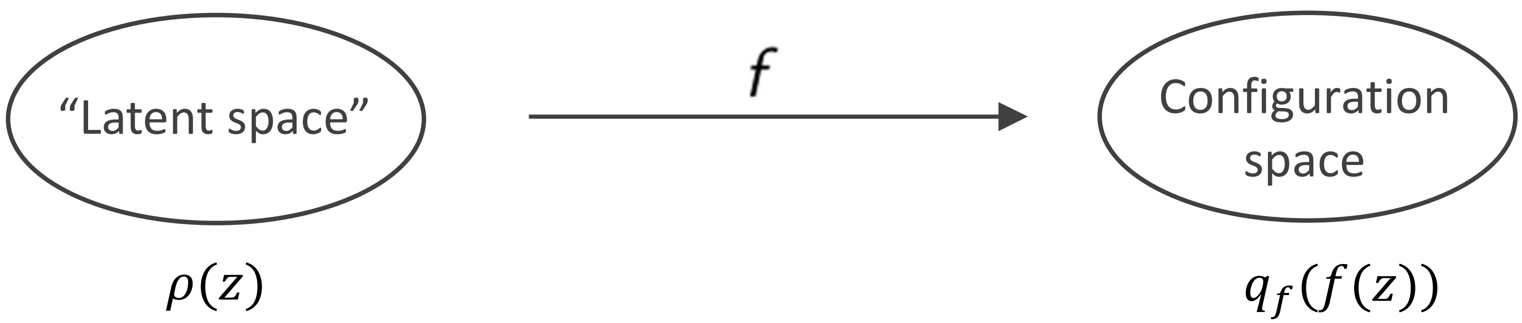
  - continuous limit: physical size $L \times a$ *fixed, lattice spacing* $a \to 0$.

  - phase transition limit: $\frac{\xi}{a} \to \infty$

Flow-Based Approach

[D. J. Rezende and S. Mohamed. Variational inference with normalizing flows. ICML 2015.]

$$f\colon \mathbb{R}^{L^D} \to \mathbb{R}^{L^D} \quad *$$
$$z \mapsto f(z) = \phi$$



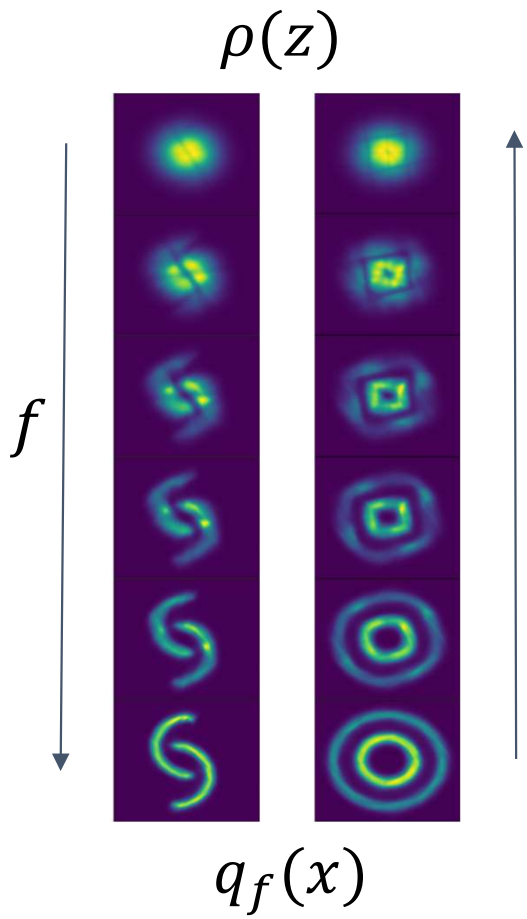
Pushed forward probability:

$$q_f\big(f(z)\big) := \rho(z) + \det\big(J_f\big)\,, \qquad (J_f)_{xy} = \frac{\partial f_y}{\partial z_x}$$

* We choose Latent space ≅ Configuration Space, and f a bijection.

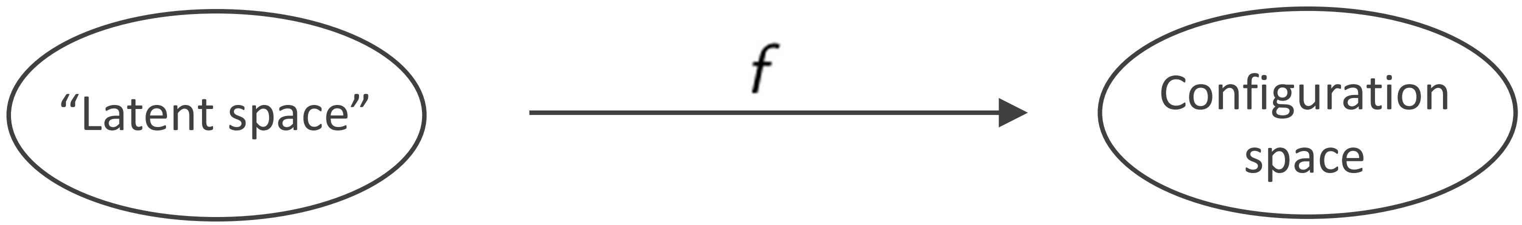
Flow-Based Approach

[D. J. Rezende and S. Mohamed. Variational inference with normalizing flows. ICML 2015.]



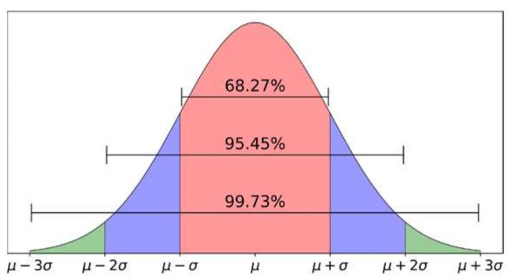
[Silde credits: Jonas Köhler]

Key to computational feasibility and efficiency



$\rho(z)$ $\qquad\qquad$ $q_f(f(z))$

1. $\rho(z)$ is easy to sample (e.g. ind Gaussian)



3. $q_f(f(z))$ approximates well the target physical distribution

$$p(\phi) := \frac{e^{-S[\phi]}}{Z}$$

Pushed forward probability:

$$q_f(f(z)) := \rho(z) / |\det(J_f)| \;, \qquad (J_f)_{xy} = \frac{\partial f_y}{\partial z_x}$$

2. $\det(J_f)$ can be easily computed/approximated

Reverse KL Training

How to make sure $q_f(f(z))$ approximate well the target physical distribution $p(\phi)$?

- Choose an Ansatz for the flow (with a computable Jacobian). Parametrise the flow with parameters $f = f_\theta$.

- Find the parameters θ that minimize the "reverse Kullback-Leibler (KL)" cost function:

$$\mathrm{KL}(q_{f_\theta}|p) = \int d\phi \; q_{f_\theta}(\phi) \log\left(\frac{q_{f_\theta}(\phi)}{p(\phi)}\right)$$

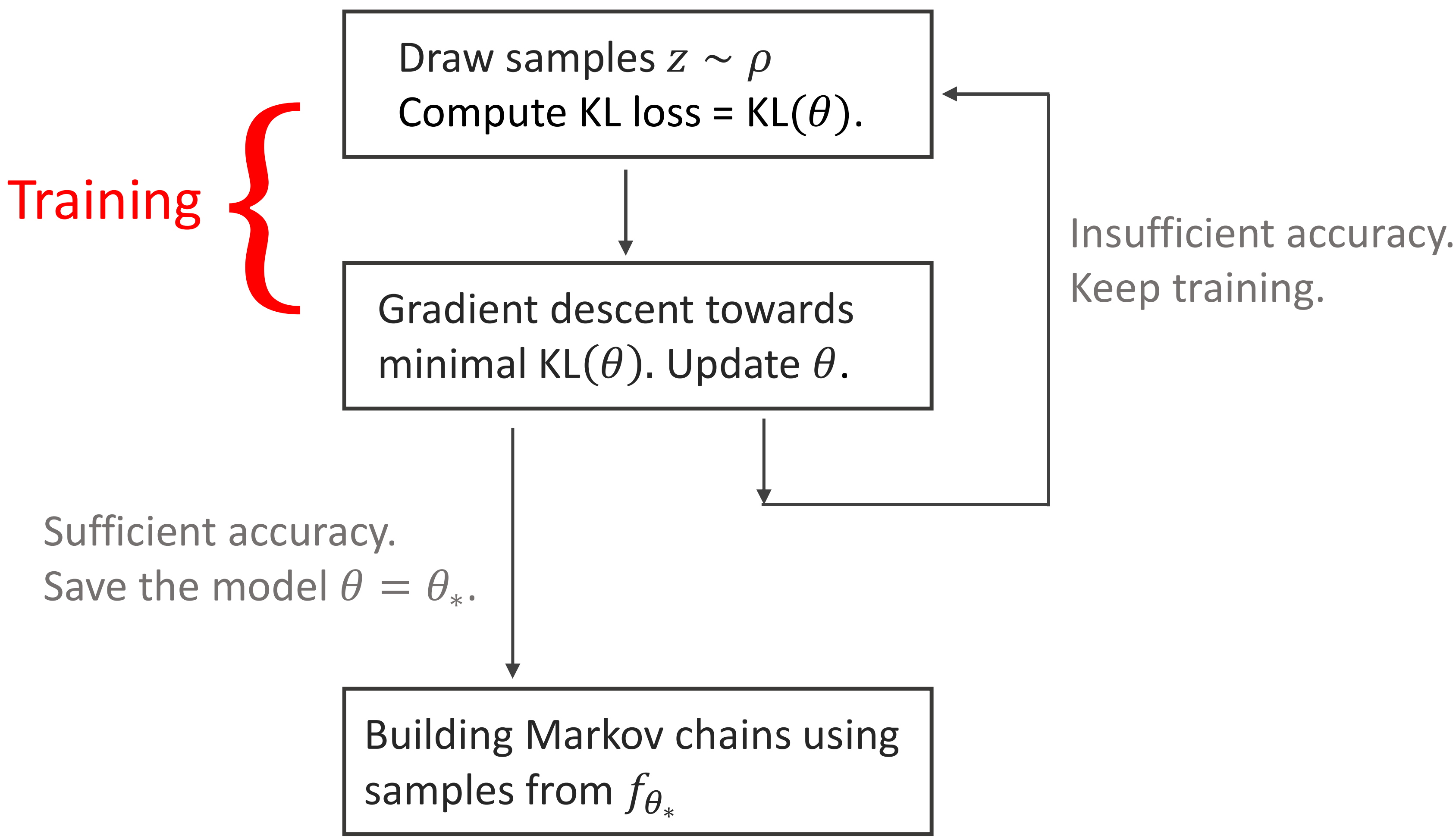
$$= \mathbb{E}_{z\sim\rho}\left[\log(q_{f_\theta}(f_\theta(z)) + S(f_\theta(z)))\right] + \mathrm{const}$$

$$p(\phi) := \frac{e^{-S[\phi]}}{Z}$$

Note: no prepared samples needed for training.

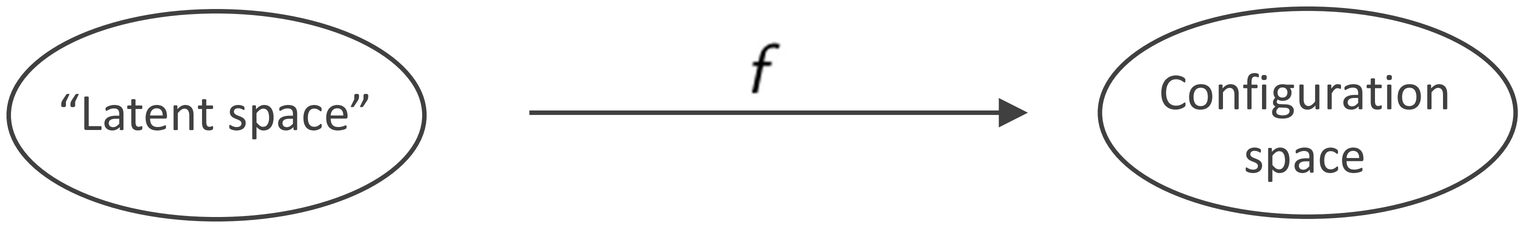
Finding $\boldsymbol{f = f_\theta}$

Given the parametrisation $f = f_\theta$,



Note: **Independent** samples (next sample ind. of the previous one).

Sampling from the model: Latent space sampling+ Metropolis-Hasting



$z \sim \rho(z)$ $\quad\mapsto\quad$ $\phi = f(z)$

Samples here **Sample configurations**

To account for the discrepancy between
the true distribution $p(\phi)$ and the approximation $q_f\big(f(z)\big)$,
accept the candidate configuration $f(z)$ as $\phi_{t+1} = f(z)$ with probability

$$\min\left(1, \frac{p(f(z))/p(\phi_t)}{q_f\big(f(z)\big)/q_f(\phi_t)}\right).$$

Theoretically, asymptotic convergence to $p(\phi)$ is guaranteed.

Checking the quality of the flow:

$$\text{Acceptance rate = average of}\left(\min\left(1, \frac{p(f(z))/p(\phi_t)}{q_f\big(f(z)\big)/q_f(\phi_t)}\right)\right).$$

- Acceptance rate = 1 when $q_f = p$.
- Long chains of rejections lead to inefficient sampling

Previous work
on using flows to sample in lattice field theories:

- Flow-based generative models for Markov chain Monte Carlo in lattice field theory. 1904.12072 (PRD)

Flow-based generative models for Markov chain Monte Carlo in lattice field theory

ϕ^4-theory in 2D

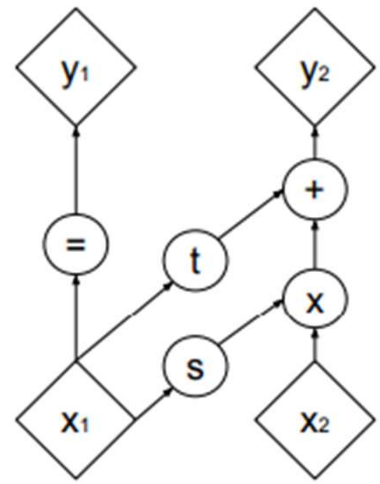
M. S. Albergo,[1,2,3] G. Kanwar,[4] and P. E. Shanahan[4,1]

Generalisations:
- Equivariant flow-based sampling for lattice gauge theory. 2003.06413 (PRL)
- Sampling using *SU*(*N*) gauge equivariant flows. 2008.05456 (PRD)
- Flow-based sampling for fermionic lattice field theories. 2106.05934
- Flow-based sampling for multimodal distributions in lattice field theory. 2107.00734

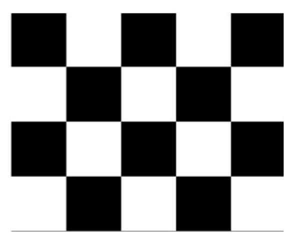
by the MIT-DeepMind-NYU group, using a discrete “*RealNVP*” flow respecting part of the lattice symmetries.

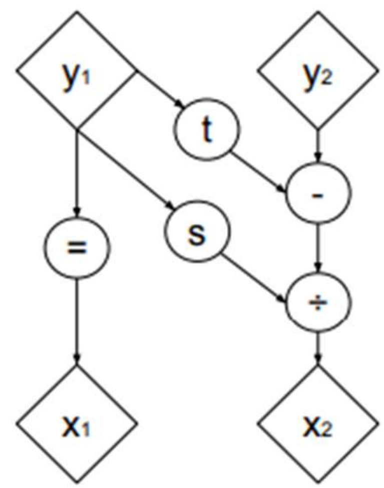
Previous work
RealNVP discrete normalizing flows

- Prior method [Albergo et al, 2018] uses realNVP [Dinh et al, 2016]
- Diffeomorphism is composed of coupling layers
  - Split input x in x_1 and x_2 of dimensions d_1 and d_2
  - $y_1 = x_1$
  - $y_2 = x_2 \odot \exp s(x_1) + t(x_1)$
  - where $s, t : \mathbb{R}^{d_1} \to \mathbb{R}^{d_2}$ parameterized neural networks
  - For example, composed of learnable linear transformations and ReLU $x \mapsto \max(0, x)$
- Inverse:
  - $x_1 = y_1$
  - $x_2 = (y_2 - t(x_1)) \odot \exp -s(x_1)$
- Easy to compute change of volume: $\log \left|\frac{\partial f(x)}{\partial x}\right| = \sum_i s(x_1)_i$
- On lattice, split is done as checkerboard, not equivariant



(a) Forward propagation

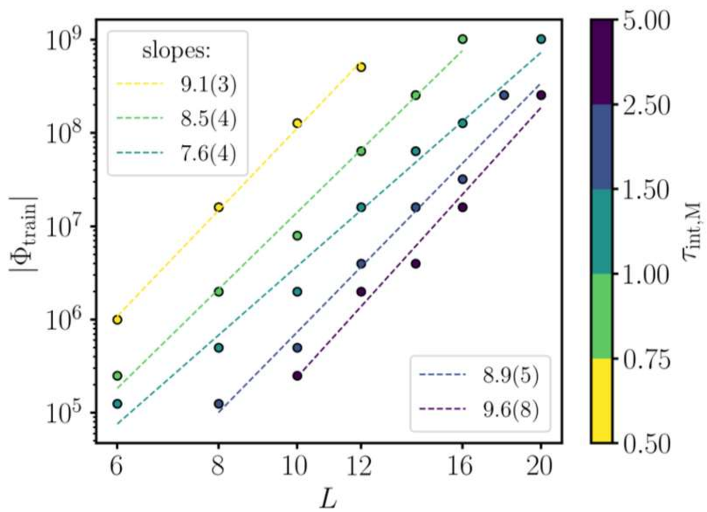




(b) Inverse propagation

Previous work
on using flows to sample in lattice field theories:

Challenges before one can scale it up: **critical slowing down of *training***
The “costs” of training the network till an acceptable acceptance rate is achieved scales very steeply with the lattice size $\sim e^{\alpha L}$.
Question: Does it really work for **large lattices**?



Efficient Modelling of Trivializing Maps for Lattice ϕ^4 Theory Using Normalizing Flows: A First Look at Scalability

Luigi Del Debbio, Joe Marsh Rossney,* and Michael Wilson
Higgs Centre for Theoretical Physics, School of Physics and Astronomy, The University of Edinburgh, Edinburgh EH9 3FD, UK

How to find a better f? Our approach: I. Symmetries

Spacetime symmetries: *e.g.* $C_L^2 \rtimes D_4$ for 2D periodic square lattices
Global symmetry: *e.g.* $\phi \mapsto -\phi$ for the ϕ^4 theory

Bake them in instead of hoping that the network will learn them.

How to find a better f? Our approach:
II. Continuous Flow (Neural ODE)

Construct the flow using an ODE:

learn this vector

$$\frac{dx(t)}{dt} = v(x(t), t; \theta)$$

$$\text{“Latent space”} \xrightarrow{f} \text{Configuration space}$$

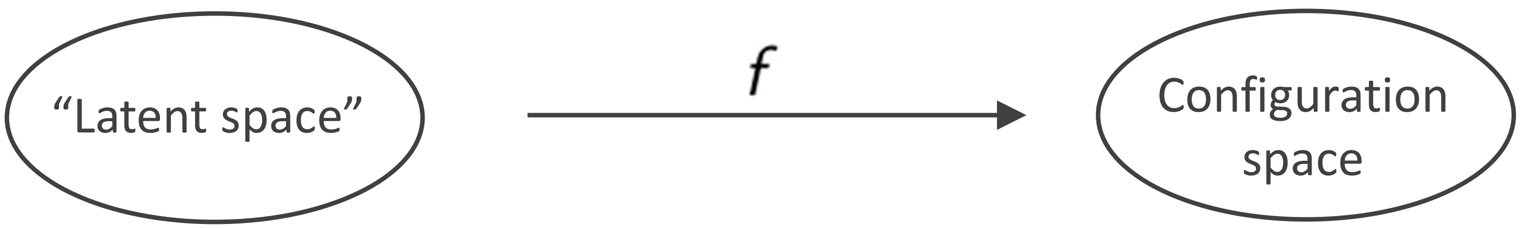
$x(0) = z$

$x(T) := f(z) = \phi$

How to find a better f? Our approach: II. Continuous Flow (Neural ODE)

3. $\det(J_f)$ can be easily computed/approximated

$$\frac{d\log p(x(t))}{dt} = -(\nabla_x \cdot v)(x(t), t; \theta)$$



$$p(x(0)) = \rho(z)$$

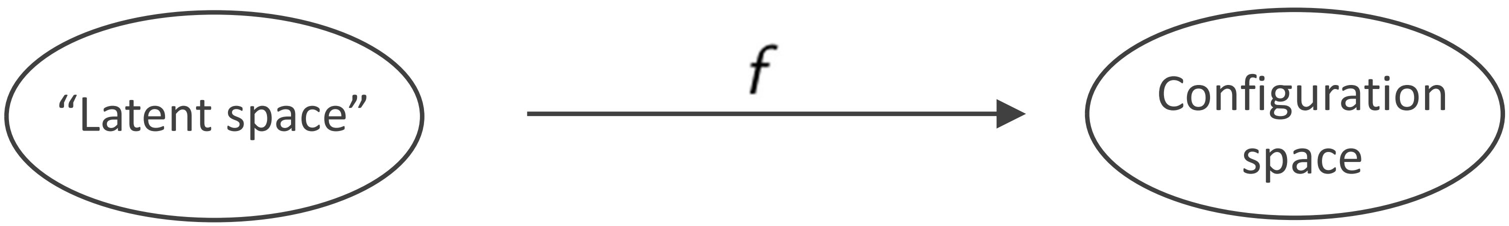
$$p(x(T)) := q_f(\phi)$$

How to find a better f? Our approach: II. Continuous Flow (Neural ODE)

[Chen, Rubanova, Betternourt, Duvenaud] NeurIPS 2018

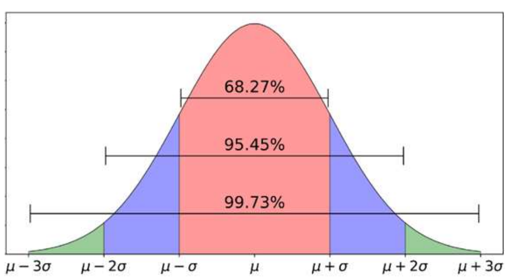
Advantages of the continuous flow:

- Flexible choices of the vector field $v(x(t), t; \theta)$.
- Modest memory cost
- Interpretability



Free theory

Interacting theory $p(\phi) := \frac{e^{-S[\phi]}}{Z}$



$$\frac{dx(t)}{dt} = v(x(t), t; \theta)$$

A learned RG flow?

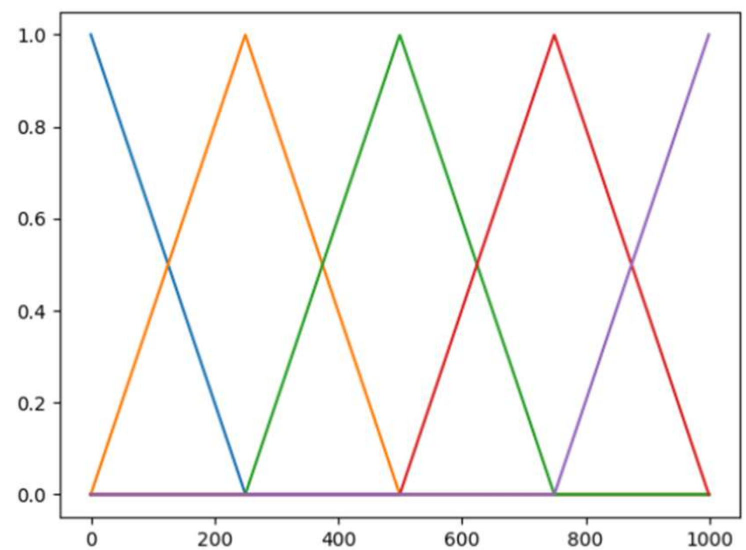
Focus on 2D ϕ^4 theory.
Our choice: a single layer shallow network

$$\frac{d\phi_i(t)}{dt} = \sum_{j,a,f} W_{i,j,a,f} K(t)_a \sin(\omega_f \phi_j(t))$$

i, j: lattice points

learnable parameters θ

$K(t)_a$: time shift kernels



Focus on 2D ϕ^4 theory.
Fully equivariant w.r.t. all symmetries.

$$\frac{d\phi_i(t)}{dt} = \sum_{j,a,f} W_{i,j,a,f}\, K(t)_a \sin(\omega_f \phi_j(t))$$

Spacetime symmetries (translation, rotation, reflection):

$$W_{i,j,a,f} = W_{|i-j|,a,f}$$

Focus on 2D ϕ^4 theory.
Fully equivariant w.r.t. all symmetries.

$$\frac{d\phi_i(t)}{dt} = \sum_{j,a,f} W_{i,j,a,f}\, K(t)_a \sin(\omega_f \phi_j(t))$$

Global symmetry:

The ODE preserves the Z_2 symmetry (no $\cos \omega\phi$ term).

So $\rho(z) = \rho(-z) \Leftrightarrow q_f(\phi) = q_f(-\phi)$.

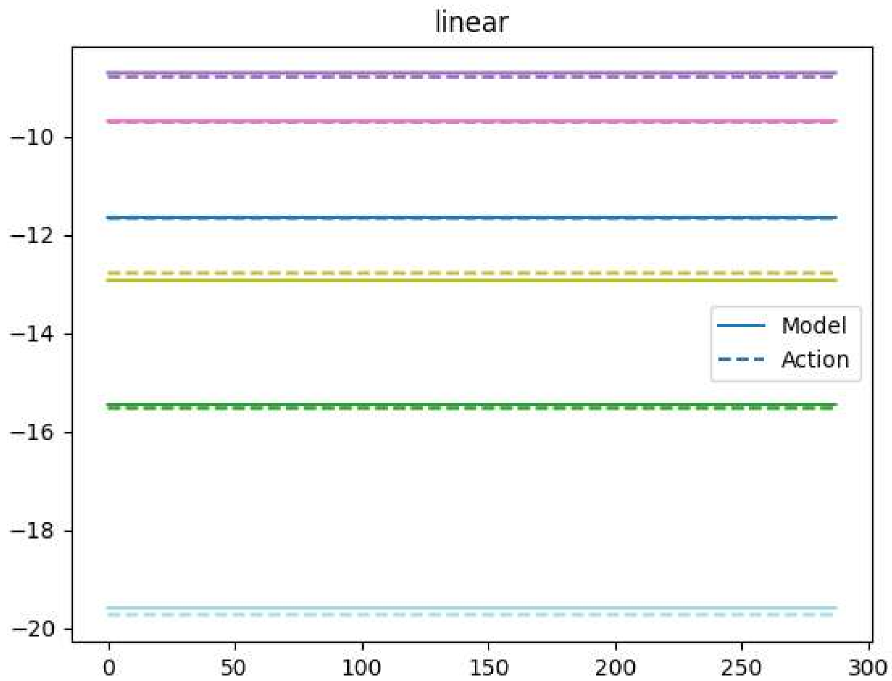
More generally, to preserve a global symmetry G, we require

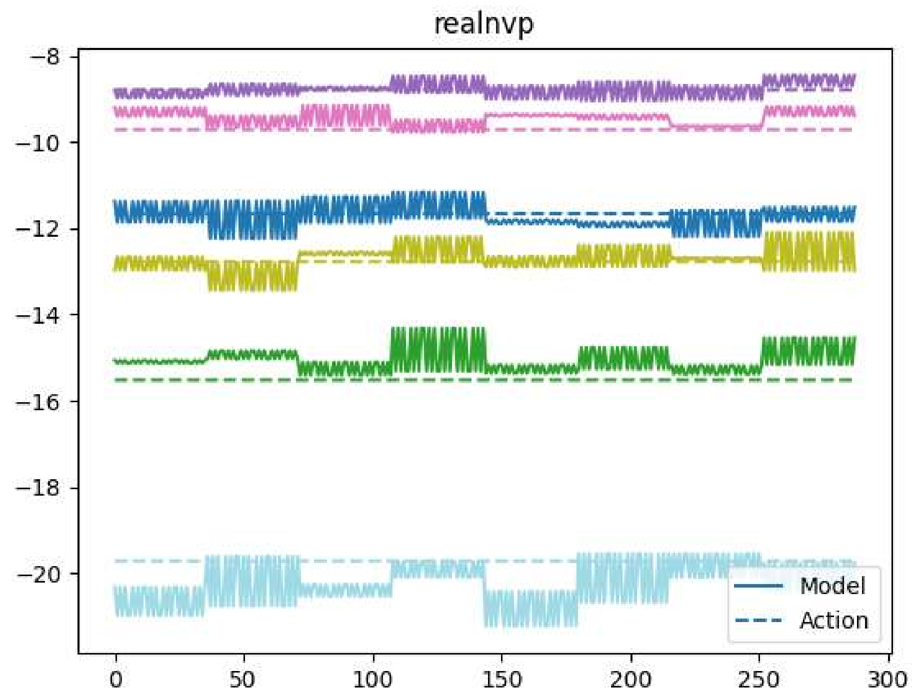
$\boldsymbol{v}(\boldsymbol{g} \cdot \boldsymbol{x}(\boldsymbol{t}), \boldsymbol{t}; \boldsymbol{\theta}) = \boldsymbol{g} \cdot \boldsymbol{v}(\boldsymbol{x}(\boldsymbol{t}), \boldsymbol{t}; \boldsymbol{\theta})$ **for all g** $\in \boldsymbol{G}$ **(equivariance)**

$e.g.\ \sin(\omega_f\,(-\phi)) = -\sin(\omega_f\,\phi)$.

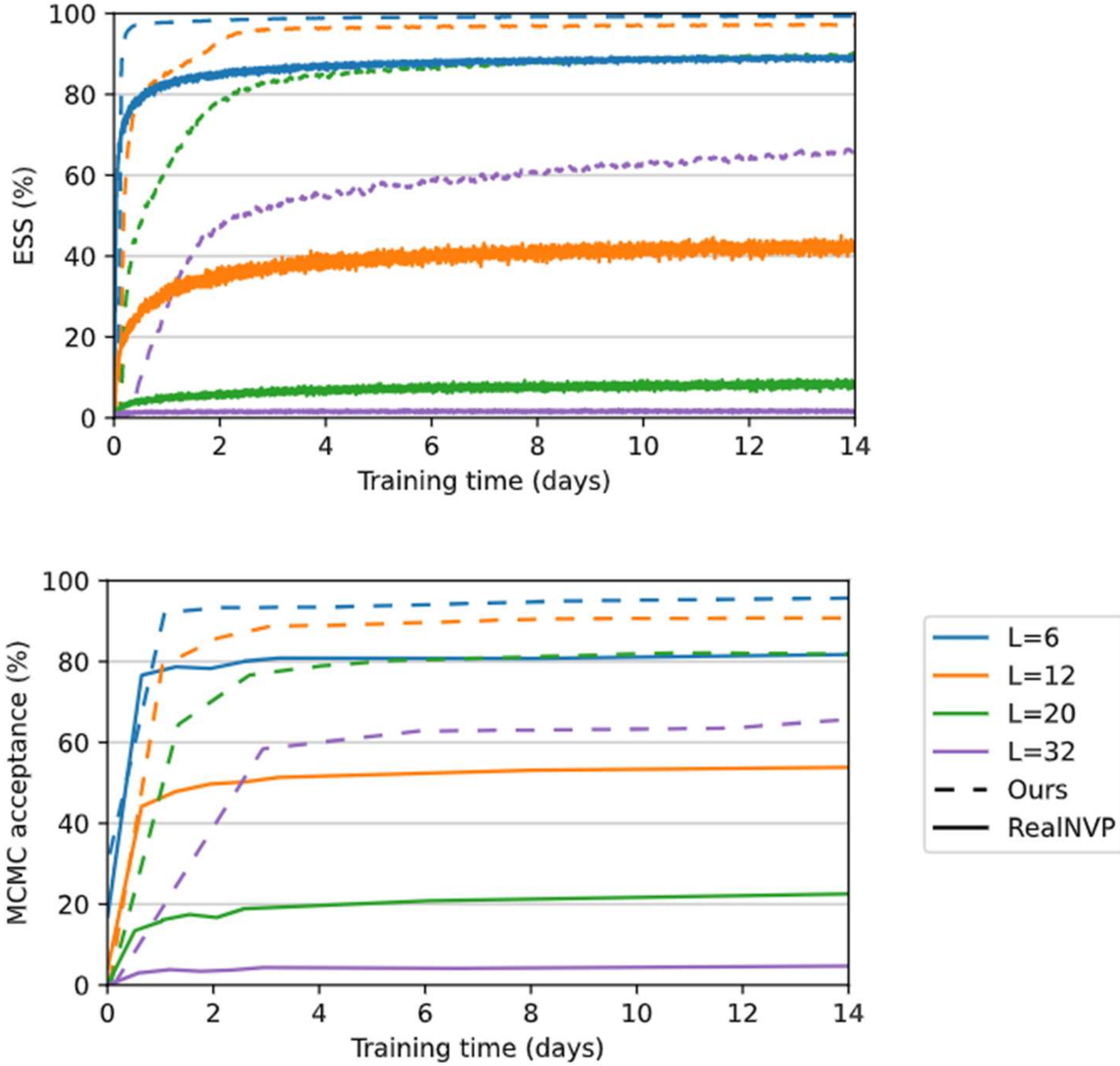
Symmetries are good.

Spacetime symmetries: better baked in





Experimental Result: drastic improvement of scalability



Lessons Learned:

- **Symmetries really help.**
  Verified by ablation study:

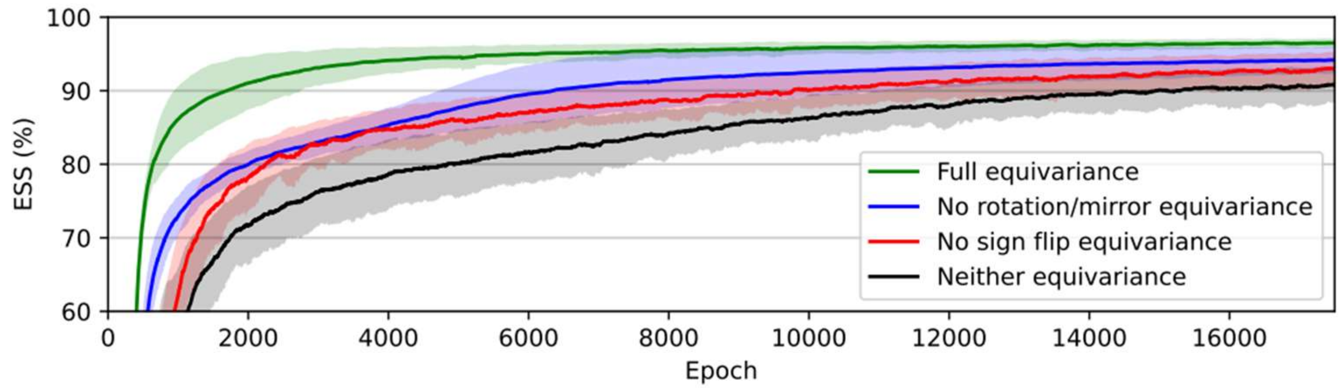


Figure 5: Ablation study on the $L = 12$ lattice, varying the equivariance properties. Shown are mean and standard deviation across three training runs per variation.

- **Deeper is not always better.**
  We used a single layer continuous flow.

Future:

- **Pushing on the physics front.**
  Goal application: LQCD. Clearly, to make a difference, scalability is a crucial must.

- **Pushing on the AI front.**
  * The complexity and the prior knowledge makes lattice field theories ideal testing grounds for new sampling technics.
  * Interpretability.