Stochastic normalizing flows for lattice field theory

Elia Cellini

Università degli Studi di Torino/Istituto Nazionale Fisica Nucleare

27th February 2023

Machine Learning approches in Lattice QCD - An interdisciplinary exchange

Based on:
M. Caselle, E. C., A. Nada., M. Panero, *JHEP* 07 (2022) 015

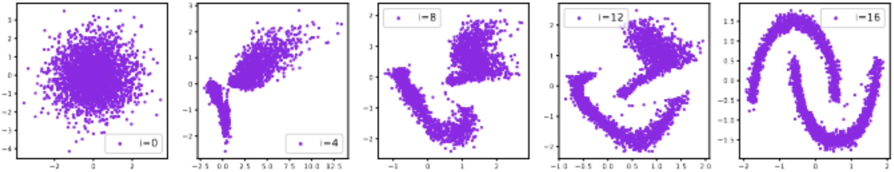
Outline

Normalizing Flows

Normalizing Flows

Normalizing flows (NFs) **[Rezende and Mohamed; 2015]** are a class of deep generative models able to learn families of compositions of diffeomorphisms (i.e. <u>invertible</u> and <u>differentiable</u> transformations):

$$\phi = \phi_N = g_\theta(\phi_0) = (g_N \circ \cdots \circ g_1)(\phi_0) \qquad y_0 \sim q_0 \qquad \theta : \textit{parameters}$$



Normalizing Flows: density estimation

The generated distribution for ϕ is

$$q_\theta(\phi) = q_N(\phi_N) = q_0(g_\theta^{-1}(\phi_N)) \prod_n |\det J_n(\phi_n)|^{-1}$$

Normalizing Flows: density estimation

The generated distribution for ϕ is

$$q_\theta(\phi) = q_N(\phi_N) = q_0(g_\theta^{-1}(\phi_N)) \prod_n |\det J_n(\phi_n)|^{-1}$$

NFs can be trained to learn a target Boltzmann distribution $p(\phi) = e^{-S[\phi]}/Z$ by minimizing the **Kullback-Leibler divergence**:

$$D_{KL}(q_\theta||p) = \int d\phi q_\theta(\phi) \log \frac{q_\theta(\phi)}{p(\phi)} = \int d\phi q_\theta(\phi) \Big[\log q_\theta(\phi) + S[\phi] + \log Z\Big]$$

Normalizing Flows: Target Partition Function

A trained NF g_θ can be used to compute directly the partition function of the target:

$$Z = \int D\phi e^{-S[\phi]} = \int D\phi q_\theta(\phi) \frac{e^{-S[\phi]}}{q_\theta(\phi)} = Z_0 \int D\phi q_\theta(\phi) \tilde{w}(\phi) \approx Z_0 \langle \tilde{w} \rangle_{\phi \sim q_\theta}$$

[Nicoli et al.; 2020]

Normalizing Flows: Target Partition Function

A trained NF g_θ can be used to compute directly the partition function of the target:

$$Z = \int D\phi e^{-S[\phi]} = \int D\phi q_\theta(\phi) \frac{e^{-S[\phi]}}{q_\theta(\phi)} = Z_0 \int D\phi q_\theta(\phi) \tilde{w}(\phi) \approx Z_0 \langle \tilde{w} \rangle_{\phi \sim q_\theta}$$

[Nicoli et al.; 2020]

Unnormalized weight:

$$\tilde{w}(\phi) = \exp\left(-\left\{S[\phi] - S_0[g_\theta^{-1}(\phi)] - Q\right\}\right) = \frac{\exp(-S[\phi])}{Z_0 q_\theta(\phi)}$$

with

$$q_\theta(\phi) = q_0(g_\theta^{-1}(\phi)) \exp(-Q) \qquad \underbrace{q_0(y_0) = \exp(-S_0[y_0])/Z_0}_{\text{e.g. normal distribution}}$$

Normalizing Flows: Observables

Observables can be computed using a reweighting procedure or a MCMC algorithm:

Reweighting

$$\langle \mathcal{O} \rangle_{\phi \sim p} = \frac{1}{\hat{Z}} \langle \mathcal{O} \tilde{w} \rangle_{\phi \sim q_\theta}$$

[Nicoli et al.; 2020]

Metropolis-Hastings

$$A(\phi^{(i-1)}, \phi') = \min\left(1, \frac{q_\theta(\phi)}{p(\phi)} \frac{p(\phi')}{q_\theta(\phi')}\right)$$

[Albergo et al.; 2019]

Normalizing Flows for Lattice Field Theory

Normalizing flows found successful application as a sampler for Lattice Field Theory.

- Ising Model: **[Li and Wang; 2018]**
- Scalar field theory: **[Albergo et al.; 2019]**,**[Nicoli et al.; 2020]**,
- Lattice gauge theory **[Kanwar et al.; 2020]**,**[Abbott et al.; 2022]**

See **[Abbott et al.; 2211.07541]** for a recent review of the field and backup slides for more references.

In the following sections, we outline how to include NF formalism in a non-equilibrium thermodynamics framework using **Stochastic Normalizing Flows** and **Jarzynski's equailt**y.

Jarzynski's equality and Stochastic Normalizing Flows

Jarzynski's equality

Very general, intriguing relation in non-equilibrium statistical mechanics.
Free-energy differences (at equilibrium) directly calculated with an average over **non-equilibrium processes [Jarzynski; 1997]**:

$$\frac{Z}{Z_0} = \left\langle \exp\left(-\frac{\mathcal{W}}{T}\right)\right\rangle_f$$

Stochastic non-equilibrium evolutions

For an MCMC:

- the stochastic **non-equilibrium** evolution starts from a configuration sampled from the initial distribution q_0 and reaches the target (final) distribution p

$$q_0 \simeq e^{-S_{\eta_0}} \xrightarrow{P_{\eta_1}} e^{-S_{\eta_1}} \xrightarrow{P_{\eta_2}} \dots \xrightarrow{P_{\eta_N}} e^{-S_{\eta_N}} \simeq p$$

- the system evolves using regular Monte Carlo updates with transition probability P_{η_n}
- η_n is a **protocol** that interpolates the parameters of the theory between q_0 and p

Stochastic non-equilibrium evolutions

For an MCMC:

- the stochastic **non-equilibrium** evolution starts from a configuration sampled from the initial distribution q_0 and reaches the target (final) distribution p

$$q_0 \simeq e^{-S_{\eta_0}} \xrightarrow{P_{\eta_1}} e^{-S_{\eta_1}} \xrightarrow{P_{\eta_2}} \dots \xrightarrow{P_{\eta_N}} e^{-S_{\eta_N}} \simeq p$$

- the system evolves using regular Monte Carlo updates with transition probability P_{η_n}
- η_n is a **protocol** that interpolates the parameters of the theory between q_0 and p

Along the process we compute the dimensionless **work**

$$W = \sum_{n=0}^{N-1} \left\{ S_{\eta_{n+1}} [\phi_n] - S_{\eta_n} [\phi_n] \right\}$$

Stochastic non-equilibrium evolutions for LFTs

Stochastic non-equilibrium simulations have been successfully applied to state-of-the-art lattice calculations:

- **[Caselle et al.; 2016]**: Interface free energy in $\mathbb{Z}_2$ gauge model.
- **[Caselle et al.; 2018]**: High precision determination of the SU(3) equation of state in $d = 3 + 1$ dimensions.
- **[Francesconi et al.; 2020]**: Determination of the renormalized Yang-Mills gauge coupling using the Schrödinger functional.

Related to Annealed Importance Sampling **[Neal; 1998]**: procedure equivalent to Jarzynski's equality. Very popular in machine learning community.

A common Framework

We realized that Jarzynski's relation is the same formula used to extract Z in NFs:

$$\frac{Z}{Z_0} = \langle \tilde{w}(\phi) \rangle_{\phi \sim q_\theta} = \langle \exp(-W) \rangle_{\text{f}}$$

General dimensionless "work" :

$$W(\phi_0, \ldots, \phi_N) = S(\phi_N) - S_0(\phi_0) - Q(\phi_1, \ldots, \phi_N) = -\ln \tilde{w}(\phi)$$

while the "heat" Q depends on the type of flow.

A common Framework

normalizing flows

$$\phi_0 \to \phi_1 = g_1(\phi_0) \to \cdots \to \phi$$

$$Q = \sum_{n=0}^{N-1} \ln \left|\det J_n(\phi_n)\right|$$

$$D_{KL}(q_\theta \| p) = -\langle \ln \tilde{w}(\phi) \rangle_{\phi \sim q_\theta} + \ln \frac{Z}{Z_0}$$

stochastic non-equilibrium evolutions

$$\phi_0 \overset{P_{\eta_1}}{\to} \phi_1 \overset{P_{\eta_2}}{\to} \dots \overset{P_{\eta_N}}{\to} \phi_N$$

$$Q = \sum_{n=0}^{N-1} S_{\eta_{n+1}}(\phi_{n+1}) - S_{\eta_{n+1}}(\phi_n)$$

$$D_{KL}(q_0 P_f \| p P_r) = \langle W \rangle_f + \ln \frac{Z}{Z_0}$$

Stochastic Normalizing Flows

Stochastic Normalizing Flows (SNFs) (introduced in **[Wu et al.; 2020]**)

$$\phi_0 \to g_1(\phi_0) \overset{P_{\eta_1}}{\to} \phi_1 \to g_2(\phi_1) \overset{P_{\eta_2}}{\to} \dots \overset{P_{\eta_N}}{\to} \phi_N = \phi$$

$$Q = \sum_{n=0}^{N-1} S_{\eta_{n+1}}(\phi_{n+1}) - S_{\eta_{n+1}}(g_n(\phi_n)) + \ln|\det J_n(\phi_n)|$$

SNF idea is similar to the CRAFT approach **[Matthews et al.; 2022]**

Testing Stochastic Normalizing Flows

SNFs for the ϕ^4 2d model

Typical toy model for tests: ϕ^4 scalar field theory in 2 dimensions

$$S(\phi) = \sum_{x \in \Lambda} -2\kappa \sum_{\mu=0,1} \phi(x)\phi(x+\hat{\mu}) + (1-2\lambda)\phi(x)^2 + \lambda\phi(x)^4$$

target parameters $\kappa = 0.2$ and $\lambda = 0.022$ (as in **[Nicoli et al.; 2020]**): unbroken symmetry phase

SNFs architecture

Stochastic evolutions

η_n interpolates between the prior (normal distribution is recovered with $\kappa = \lambda = 0$) and target parameters

- linear protocol η_n
- heatbath algorithm for the stochastic updates
- $n_{sb} = \#$ of stochastic updates

Normalizing Flow

- $g_i =$ affine coupling layers as described in the **RealNVP** architecture **[Dinh et al.; 2016]**
- each coupling layer has two convolutional kernels with 3×3 kernel and 1 feature map
- Affine block = odd coupling layer + even coupling layer
- $n_{ab} = \#$ of affine blocks

Simple PyTorch implementation: github.com/eliacellini/SNF_for_LFT.

Testing SNFs

Goals

- can we train SNFs efficiently?
- can we improve both on NFs and on stochastic evolutions?

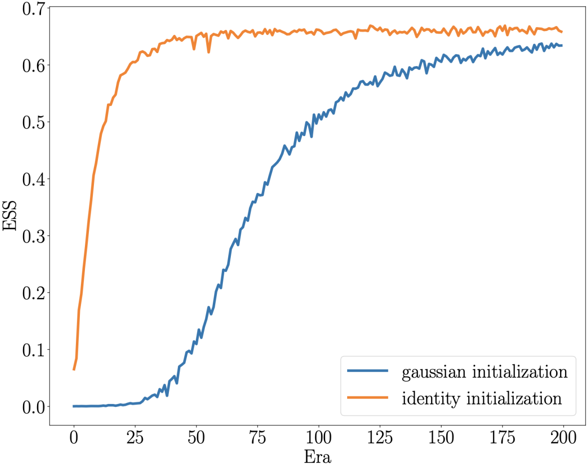
Using the Effective Sample Size as metric to evaluate architectures

$$\mathrm{ESS} = \frac{\langle \tilde{w} \rangle_{\mathrm{f}}^2}{\langle \tilde{w}^2 \rangle_{\mathrm{f}}}$$

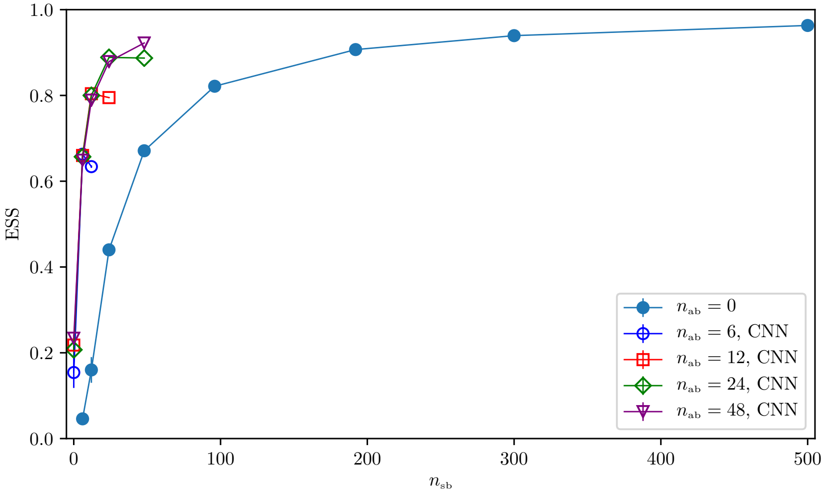
$\mathrm{ESS} = 1 \rightarrow$ perfect training

Learning curve of SNF

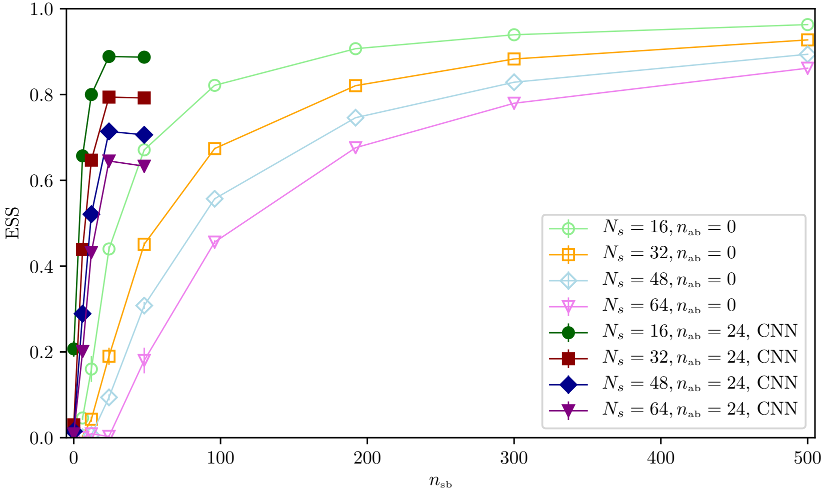
Identity initialization for NFs layers implies that the untrained model reduces to the underlying non-equilibrium algorithms **[Matthews et al.; 2022]**. 1 Era=10 gradient updates.



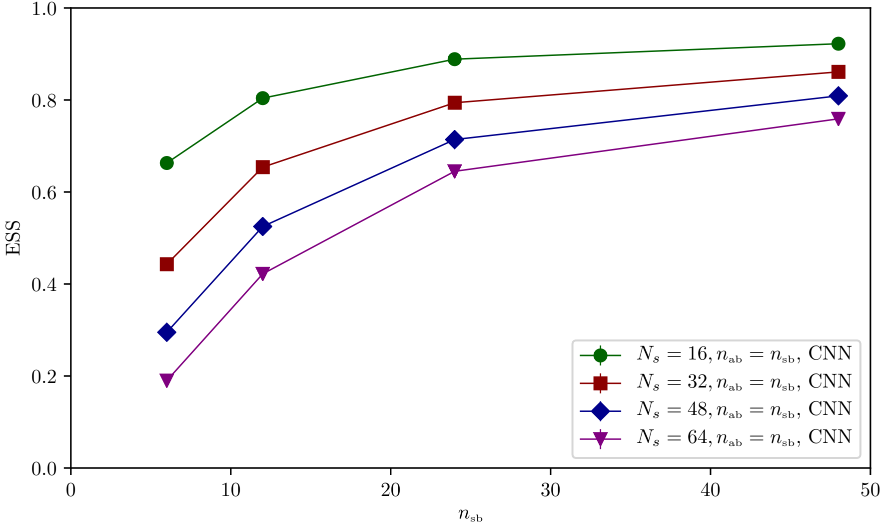
Comparing stochastic evolutions with (S)NFs on a $N_s \times N_t = 16 \times 8$ lattice,



Training length: 10^4 epochs for all volumes. Slowly-improving regime reached fast



ESS comparison for Ns×8 lattices between CNN-based SNFs with $n_{sb} = n_{ab}$, $N_t = 8$ at $\kappa = 0.2$ and $\lambda = 0.022$.



Conclusion

Conclusions

The common framework between Jarzynski's equality and NFs is now explicit

General idea: use knowledge from non-equilibrium SM to create efficient SNFs

- Example: exploit fluctuation relations to build efficient loss.

Conclusions

The common framework between Jarzynski's equality and NFs is now explicit

General idea: use knowledge from non-equilibrium SM to create efficient SNFs

- Example: exploit fluctuation relations to build efficient loss.

SNFs

- SNFs with CNNs and $n_{sb} = n_{ab}$ have a promising volume scaling at fixed training length

Conclusions

The common framework between Jarzynski's equality and NFs is now explicit

General idea: use knowledge from non-equilibrium SM to create efficient SNFs

- Example: exploit fluctuation relations to build efficient loss.

SNFs

- SNFs with CNNs and $n_{sb} = n_{ab}$ have a promising volume scaling at fixed training length

SNFs vs. stochastic evolutions

- SNFs might be an even better method!
- trade-off: training for less MCMC updates

Conclusions

The common framework between Jarzynski's equality and NFs is now explicit

General idea: use knowledge from non-equilibrium SM to create efficient SNFs

- Example: exploit fluctuation relations to build efficient loss.

SNFs

- SNFs with CNNs and $n_{sb} = n_{ab}$ have a promising volume scaling at fixed training length

SNFs vs. stochastic evolutions

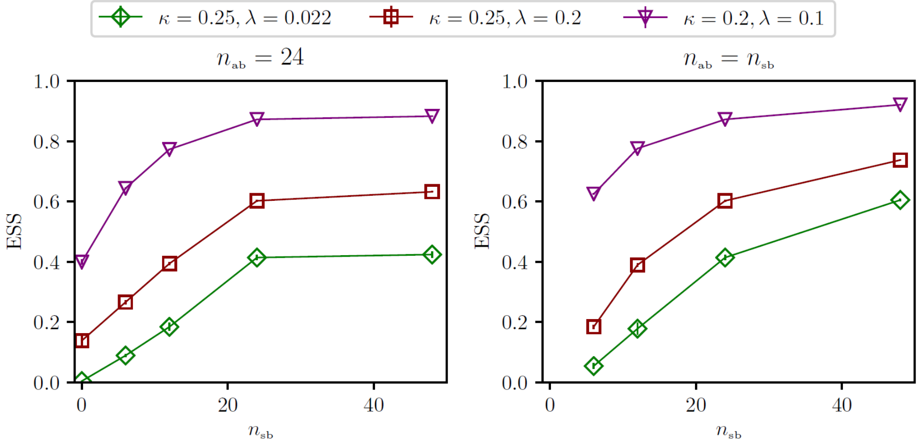
- SNFs might be an even better method!
- trade-off: training for less MCMC updates

SNFs vs. normalizing flows

- improve scalability ?
- improve interpretability

Thank you for your attention!

Test with different action parameters (unbroken symmetry phase) on a $N_s \times N_t = 16 \times 8$ lattice



Related works

- Annealed Importance Sampling **[Neal; 1998]**: procedure equivalent to JE. Very popular in ML community. Used in SNF paper **[Wu et al.; 2020]**
- AIS $\rightarrow$ generalized in Sequential Monte Carlo (SMC) samplers **[Dai et al.; 2020]**. Also well known in Machine Learning.
- SNF idea reworked in CRAFT approach **[Matthews et al.; 2022]**
- **[Vaikuntanathan and Jarzynski; 2011]**: related approach with deterministic mappings on top of non-equilibrium transformations. No neural networks.
- **[Midgley et al.; 2022]**: Combination of NFs and AIS.

Normalizing Flows for LFT, references:

- **[Li and Wang; 1802.02840]**: Ising model simulations (2018)
- **[Albergo et al.; 1904.12072]**: Flow-based generative model for MCMC in scalar lattice field theory (2019).
- **[Nicoli et al.; 1910.13496]**: Computing thermodynamics observables with NFs (2020).
- **[Kanwar et al.; 2003.06413]**: Equivariant NFs for Lattice Gauge Theory (2020).
- **[Boyda et al.; 2008.05456]**: Gauge equivariant flows for $SU(N)$ gauge theories (2020).
- **[Albergo et al.; 2101.08176]**: Introduction to NFs for Lattice Field Theory with code in PyTorch (2021).
- **[Del Debbio et al.; 2105.12481]**: First systematic study of the scaling of NFs (2021).
- **[Hackett et al.; 2107.00734]**: Flow based sampling for multimodal distributions (2021).

Normalizing Flows for LFT, references:

- **[Lawrence et al.; 2101.05755]** Normalizing Flows for the real-time sign problem (2021).
- **[Albergo et al.; 2106.05934]**: Flow-based sampling for fermionic lattice field theories (2021).
- **[de Haan et al.; 2110.02673]** and **[Gerdes et al.; 2207.00283]**: Equivariant Continuous NFs for Lattice Field Theory (2021/2022).
- **[Finkenrath; 2201.02216]**: Tackling critical slowing down using global correction steps with equivariant flows (2022).
- **[Albergo et al.; 2202.11712]**: Flow-based sampling in the lattice Schwinger model (2022).
- **[Pawlowski and Urban; 2203.01243]**: Flow-based density of states for complex actions (2022).
- **[Singha et al.; 2207.00980]**: Conditional Normalizing flow for lattice scalar field theory (2022).
- **[Vaitl et al.; 2206.09016]**: Path Gradient Estimator for Continuous NFs (2022).

Normalizing Flows for LFT, references:

- **[Abbott et al.; 2207.08945]**: Gauge-equivariant flow models for sampling in lattice field theories with pseudofermions (2022).
- **[Abbott et al.; 2211.07541]**: Aspects of scaling and scalability for flow-based sampling of lattice QCD (2022).
- **[Chen; 2211.03470]**: Fourier-Flow model for generating Feynman paths (2022).
- **[Bacchio et al.; 2212.08469]**: Learning Trivializing Gradient Flows for Lattice Gauge Theories (2022).
- **[Albandea et al.; 2302.08408]**: Learning Trivializing Flows (2023).